

# **Solving software maintenance challenges using a Lean Service Creation approach**

Loredana Tauciuc

<b>Author(s)</b> Loredana Tauciuc	
<b>Degree programme</b> Information Systems Management	
<b>Thesis title</b> Solving software maintenance challenges using a Lean Service Creation approach	<b>Number of pages and appendix pages</b> 87 + 21
<p>The software development world has been changing constantly, and with it, the paradigms used for software implementation and management. If traditionally a sequential approach was favored, today a more holistic approach is embraced. The value of inter-skilled teams is being proven every day, and the cycle for creating software services is much faster, while the users have gained a more central role.</p> <p>While these new paradigms have focused on creating new services, the study is focusing on the after-launch period of a service. Traditionally, this period was known as software maintenance. However, the scope of this study is to understand how the tools developed for new service creation can help the teams doing software maintenance. Future and its Lean Service Creation mindset are used as a backdrop for the study. Based on this setting, the theoretical framework covers concepts such as: lean startup, agile development, design thinking, software continuous development, Teal and chaordic organizations, software maintenance as software continuous development.</p> <p>During spring-winter 2017, several experiments to alleviate the problems of the continuous development team are being ran. All the experiments have been based on the Lean Service Creation mindset. Their impact has been assessed with periodical surveys, and one focus group.</p> <p>The study shows that tools such as Lean Service Creation planning canvases, visual time usage radiator, as well as practices such as working in sprints, using story points or having cross-teams daily or weekly discussions are valuable and have a positive impact on the teams doing continuous development. These findings can be applied universally, including to teams that want to have an approach closer to the Lean Service Creation approach in the creation and further development of their services.</p>	
<b>Keywords</b> software maintenance, lean service creation, continuous development, design thinking, growth hacking, continuous development team, software development team challenges, lean startup, design thinking, agile development, chaordic organizations, teal organizations	

## Table of contents

1	Introduction .....	5
1.1	Why should we look at continuous development? .....	5
1.2	The context of the study – a view on Futurice .....	6
1.3	Current challenges in the context of the study .....	6
1.4	The objectives of the thesis .....	7
1.5	The research problem .....	8
1.6	How the thesis is structured .....	9
2	Theoretical framework .....	11
2.1	Agile software development .....	11
2.2	Lean startup .....	15
2.3	Design thinking and service design .....	18
2.4	Lean Service Creation .....	25
2.5	Continuous software development and software maintenance .....	29
2.6	New generation of organizations .....	32
2.7	Premises for the study .....	36
2.7.1	The Culture: is there buy in? .....	37
2.7.2	The Team: is it suitable and ready for LSC? .....	38
2.7.3	The Work: does it permit LSC? .....	40
3	Research design .....	41
3.1	Study setup .....	41
3.1.1	Target and objective of the study .....	41
3.1.2	The continuous software development team's setup .....	42
3.1.3	The author's role in the study .....	43
3.1.4	Defining the focus of the research question: what are the main problem areas to be addressed? .....	43
3.2	Data collection .....	45
3.2.1	Data collection methods .....	46
3.2.2	The basis for choosing the experiments .....	47
3.2.3	Overview of experiments chosen .....	50
3.2.4	Business objective canvas experiment .....	51
3.2.5	Focus canvas experiment .....	54
3.2.6	Service manager in case team .....	57
3.2.7	Work in sprints .....	59
3.2.8	Story points, velocity .....	60
3.2.9	Time radiator .....	61
3.2.10	Cross-teams daily scrums .....	62
3.2.11	Whole team weeklies .....	63

3.2.12 “A star and a wish” .....	64
3.3 Data analysis .....	65
3.3.1 Data analysis methods .....	65
3.3.2 Team survey analysis 2017 – 2018 .....	68
3.3.3 Team retro 2017 analysis .....	72
3.3.4 OfficeVibe analysis .....	74
3.3.5 How relevant has the study been to the research objectives? .....	76
4 Results .....	79
4.1 Work with customers .....	79
4.2 Visibility of work progress .....	80
4.3 Team collaboration and cohesion .....	80
5 Conclusions .....	82
5.1 Current state .....	82
5.2 Recommended next steps .....	83
6 Discussion .....	84
References .....	86
Appendices .....	1
Appendix 1 Agile Manifesto .....	1
Appendix 2 Team retro workshop notes, 2017 .....	2
Appendix 3 Team retro workshop notes, sorted, 2017 .....	3
Appendix 4 Interview structure for interview with Hanno Nevanlinna .....	4
Appendix 5 Interview with Hanno Nevanlinna, co-creator of LSC. ....	5
Appendix 6 Team survey invitation .....	8
Appendix 7 Team survey questions, June 2017 .....	9
Appendix 8 Team survey questions, August 2017 .....	10
Appendix 9 Team survey questions, January 2018 .....	11
Appendix 10 Team survey January 2018, Part 2 answers .....	13
Appendix 11 Team focus group, January 2018 – notes .....	21

# 1 Introduction

In the digital age, software is not an unusual commodity anymore. There are companies for which software and software services are the only product they sell, and there are more traditional companies that need and use software to enhance their core business (for example, having a webstore for selling physical goods). Because we are at the confluence of these two worlds (software oriented and more traditional), there needs to be a common language that the two extremes can use to understand each other and cooperate. Part of the common language is defining what a digital solution or digital service is and understanding its lifecycle.

## 1.1 Why should we look at continuous development?

The lifecycle of a digital service is similar to the lifecycle of a car, and acquiring a digital service implies the same attitude and actions that acquiring a car implies. When a customer buys a car, the journey only begins. For the customer to use the car and go to places, the customer needs to refill the tank, keep the car clean, do the inspections. To make the car even more comfortable, the owner brings own CDs, installs a child seat when the family grows, installs extra storage for the winter sports equipment, etc. The owner adapts the car's features to her changing needs, so that the car keeps relevant.

A digital service works in a similar fashion. While creating a new service is how the digital world goes around, all the services created need to be able to serve the user once they are launched. For this to happen, the company responsible for the service needs to constantly listen to the users, understand their needs, and adapt their service so that it keeps on satisfying the needs of the users. This way, the digital service keeps relevant to the users, and in the same time goes on fulfilling the business objectives the owner company set.

So, for the service to keep relevant to both its users and the business, it needs to be continuously developed. Nowadays, there is a not-so-rare common misconception that a service is "done" once it is launched. Today, when speed is important when launching a digital service, the focus is on getting feedback early from the users, so the launched version is often far from being complete. This is one more reason to keep on developing it after it becomes available for the users. Otherwise, the digital service will become obsolete, as no one will use it.

## **1.2 The context of the study – a view on Futurice**

The study that is referred to in this work has been conducted in Futurice. Futurice is building digital services and is, according to the website, “built on trust”. Originally started in Helsinki, it now boasts more than 450 employees in 6 different offices in 4 countries and has won the *Best place to work* award in Europe in 2012 and 2013.

The three building blocks for Futurice’s philosophy are described in the 3x2 model, a tool that empowers decision making, transparency and fairness. When making any decision, an employee needs to consider 3x2 aspects: how does the decision affect the colleagues, the customer and the numbers, now and in the future. In practice, it means that the company has created a system where the employees, the customers and the numbers are all important.

The 3x2 model together with the values and the way of working promoted inside Futurice create an environment where employees are invited and encouraged to self-manage. They are also seen as wholesome beings and are invited to participate in setting up the purpose of the company and the vision. These are characteristics of a new generation of organizations, called TEAL organizations (Laloux, 2014).

## **1.3 Current challenges in the context of the study**

Futurice is not only building digital services. The company is also the creator of Lean Service Creation (LSC), a toolkit that brings structure and focus to service creation. The toolkit is open source, available online, and it can be taken into use and modified by anyone. It has a wide range of applicability, from digital services to museums. The main concept in LSC is the build-measure-learn mindset. We will discuss more about LSC a following chapter.

Naturally, in a company where the continuous improvement mindset is ingrained in how people are working, everyone understands that the actual information about how the service is serving its users comes only when the users start accessing the service. We live in the age of data, and real data is available only when the services are in real use. Once the service is launched, it can and should be adjusted and modified further. This way, our customers can get more out of their services and can serve their own customers better. Services are living organism.

LSC defines a service lifecycle as having two phases: the pre-minimum viable product (pre-MVP), and the post-MVP. In the pre-MVP phase, a service is designed and it’s MVP

is built and launched. During post-MVP, the service is further developed, grown and kept relevant.

Currently, the accent has been on the pre-MVP phase, as Futurice's customers have been typically looking for new interesting services and so the company has specialized in launching new minimum lovable products. It has positioned itself as a "cool kid doing cool services", according to one of the company's sales people.

Inside the company, this lead to the creation of two employee groups: the ones working on the new services (pre-MVP), and the ones working on keeping the existing services relevant (post-MVP). The perception on the two groups is different, the pre-MVP group being the one doing the interesting work, whereas the post-MVP group is seen as doing basic maintenance work for services that do not have any potential anymore. Having this discrepancy comes against the satisfaction and fairness promise that Futurice makes its employees. Also, the build-measure-learn cycle is broken, as the company is not actively helping the customers keeping their service relevant – not in the same way as pre-MVP.

In order to ensure high work satisfaction for all its employees, Futurice needs to align the whole service lifecycle to the build-measure-learn principle and find a fresh approach to continuous development.

#### **1.4 The objectives of the thesis**

As mentioned in the previous chapter, the lifecycle of a digital service can be considered as being composed of two parts: the pre-MVP launch, and the post-MVP launch. While LSC incorporates both phases in its approach, at the practical level these two phases face a slight disconnection. When a service is launched, it is in an MVP state, and it requires continuous development and improvement so that the service keeps relevant to the target users. That is exactly the principle this study is empowering, by looking at how to apply this idea in practice. Because of this, this research has found the following objectives as being crucial for this study:

1. Understand what are some of the big challenges that the teams doing continuous development are currently facing
2. Investigate how these challenges could be tackled with a LSC approach
3. Create tools to implement the suggested solutions, possibly adding them to the LSC toolbox as extensions
4. Validate the above-mentioned solutions
5. Provide an approach that can help sustain innovation over time

In Futurice, we are looking at actively supporting our customers in keeping the services relevant and growing them after they are launched. We are not working as a team who

only reacts to issues reported. So, from now on, the author will use the term “software continuous development” or “post-MVP work”. This will refer to continuously implementing new features to an already launched service, with the scope of keeping the service relevant or growing the service in relation to the business objectives.

## **1.5 The research problem**

This research project focuses on helping to bring the work of the dedicated continuous development team in the same framework as the work of the teams creating new services. Because of that, the most relevant research question for this study can be summarized as following:

How can LSC tools help developers overcome their problems during the software continuous development phase?



## 1.6 How the thesis is structured

This project is unifying different concepts, such as software implementation and management paradigms, and focuses their applicability towards solving a focused range of challenges that the continuous development team inside Futurice has. The concepts are chosen to reflect the backdrop of the study: the management concepts that Futurice is driving (teal and chaordic organizations) and the Lean Service Creation mindset (including the concepts it is based upon, such as lean startup, design thinking and agile development). Because of this, this study begins with defining the theoretical concepts used throughout the research, then it continues to find a narrower focus on the problems to be tackled. Once the problems are identified, several solutions are proposed and the piloted as experiments. All experiments are explained in detail, together with the reasoning for choosing them. In the end, the impact of the experiments is assessed against the original problem areas and study objectives, and conclusions are drawn. Based on this, the report consists of the following chapters:

- 1 **Theoretical framework:** it introduces the main theoretical concepts, such as software maintenance, Lean Service Creation and its building blocks (agile, lean startup, design thinking), and some concepts about a new generation of organizations. It also discusses the premises of the study, where the concepts mentioned above are integrated to provide a starting point for the question: How can LSC, together with this building blocks and the principles of new generation of organizations be relevant for software continuous development in the context of Futurice.
- 2 **Research design:** it covers the setup of the study, using concepts from lean and design thinking. During the setup of the study, the major problem areas are identified using prior theoretical and empirical research. The chapter also explains how and what kind of data was collected, presenting all the solution suggestions that have been piloted during the study. The chapter closes with the methods used for the data analysis.
- 3 **Results:** this chapter looks at the impact of the experiments done in the data collection phase, correlating them to the problem areas identified. It presents metrics, together with the author's view from the three roles that the author had had during the study (see following chapter for more information).
- 4 **Conclusions:** the chapter offers a summary on the results and their impact and how they shape the current state of the research question. It also presents recommended next steps based on the study or theoretical research.

- 5 **Discussion:** takes further views on possible next directions for the study and presents personal thoughts on how the study was conducted, including lessons learnt.

## 2 Theoretical framework

This study is based on the following main concepts: LSC, as Lean Service Creation (with its building blocks: agile development, lean and design thinking), continuous software development and concepts about the new generation of organizations.

The purpose of this chapter is to define the concepts and see what role they play in the landscape of the study, from the point of view of the development team. The context used is the research question summarized in the figure below:



Figure 1. The summary of the research question (Tauciuc, 2018)

This chapter does not try to have an extensive definition for each term, but rather presents the concepts using the filter of the equation from Figure 1.

### 2.1 Agile software development

Agile software development is one of the building blocks of LSC:



Figure 2. The research question, focus on Agile Development (Tauciuc, 2018)

Next, we will look at what agile means and what are its main principles.

Agile software development is about finding better ways of developing software. It brings a shift from the traditional software development (also known as the waterfall model), towards an iterative model for software development. In an iterative model, the software product is developed in time boxed increments, and at the end of each increment the quality of the product must be ready for release.

Agile development also changes the focus towards people, interactions, collaborations, customers and working software. It may be perceived as less structured than the waterfall model, and it aims towards adapting, building trust between individuals working together and with the customer. These principles are steaming from the Agile manifesto, that states that the following are valued and more favoured (Agile Manifesto, 2001):

“

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan”

Agile doesn't deny the value that processes, documentation, contracts and plans bring, it just puts into focus the human interaction, be it between team members, with the customers, or with the changes that are bound to appear in a software project.

To get a wider understanding of what Agile stands for, it is worth mentioning the twelve principles behind the Agile Manifesto (<http://agilemanifesto.org/principles.html> ). The Agile Manifesto can be seen in Appendix 1. It states that iterative delivery is important, continuous change is part of the development process, and it emphasizes the importance of motivated team members, their collaboration and their continuous learning.

The Agile practice guide (PMI, 2017) translates the above principles into practical information. It clarifies from the beginning that Agile understands there are two types of work: definable work and high-uncertainty work.

Definable work is seen as having clear procedures, the result is predictable as the exact problem, or a similar one, has been resolved in the past. Usually definable work is found in the production environment, where the solutions to the problems have been perfected over time and where automation is used.

“On the other hand, the work of software engineers is seen as high-uncertainty work, as it deals with new design, problem solving and not-done-before work, which is exploratory. The high-uncertainty projects have high rates of change, complexity and risk, and this

brings challenges where traditional predictive approaches are aimed to be used.” (PMI, 2017).

In our context, it means that, if at the start of the software project the product owner has a fairly good idea on the purpose of the service and the features that need to be implemented (the problem), by the end of the project the “software maintenance” and “continuous development” notions are more charged with uncertainty. This leads to situations where the customers do not know what to expect or what kind of work they will receive as continuous development. From this point of view, both the situation of creating a new software service and the situation of developing further an existing one are high-uncertainty situations that benefit from an Agile approach.

Agile development emphasized the team collaboration and team work and eliminates the role of a traditional project manager. However, Agile comes with a lightweight agile project management method, called scrum. According to Scrum Alliance, Scrum is an Agile framework for completing complex projects. Given the fact that software development is a high-uncertainty work, scrum brings useful components in approaching the development.

## SCRUM FRAMEWORK

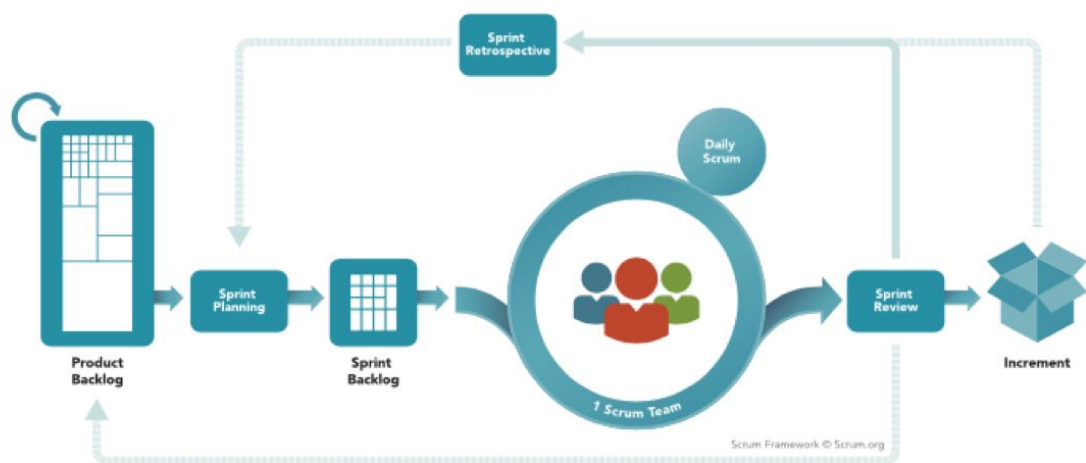


Figure 3. The scrum framework (<https://www.scrum.org>, 2018)

The main characteristics of scrum are (Leffingwell, 2007):

1. Small, cross-functional teams who work closely in sprints (30-day increments)
2. Teams are self-directed and empowered to meet the objectives of the sprints
3. Team work is facilitated by a scrum master – who eliminates impediments and reinforces the core disciplines of scrum
4. Work is organized via a product backlog, that is reprioritized for each sprint

The toolbox of Agile development contains many practices. Below I will summarize only practices relevant for this study (PMI, 2017):

- 1 Retrospectives, where the team meets and learn about, improve and adapt its processes. They happen at different key times, for e.g. when a release is completed, or any other milestone is reached
- 2 Backlog preparation, where the team and the product owner (PO) discuss and organize the list of all the work to be done. The PO has a central role in preparing the backlog, as it is tightly related to the roadmap. In the context of post-service launch, the roadmap is usually very loose, so the backlog may not be as well prepared as before the launch of the service
- 3 Backlog refinement, where work from the backlog is selected for the next iteration of the service, also called a sprint. This is the work that will be implemented next. The refinement comes from the PO and the team alike, as together they can make a more educated decision on what is important from the service's users, business and technical points of view.
- 4 Daily standups, quick meetings that happen every day, where each team member talks about what she has completed since the last standup, what she plans to complete until the next standup and what impediments she faces. These daily standups are also known as daily scrum meetings.
- 5 Demonstrations / reviews, where the team shows the working product to the customer. Usually they are held just at the end of a sprint.

The figure below illustrates how all these practices work together in the context of scrum:

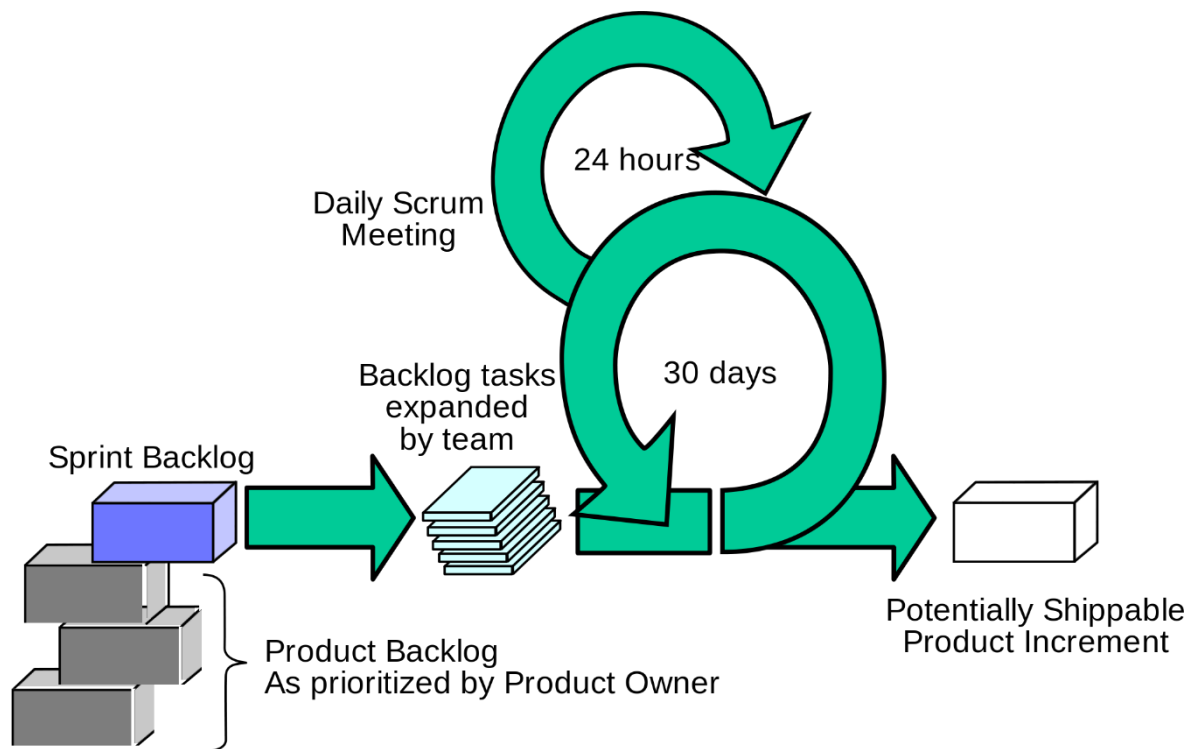


Figure 4. The general process of Agile development (Schwaber & Beedle, 2001)

The agile values and principles mentioned in this chapter can be found also in the LSC mindset. Because of this, these principles are important, as they will be incorporated in the tools investigated in the next chapters to solve the research question of this study.

## 2.2 Lean startup

Lean startup is the second building block of LSC:



Figure 5. The research question, focus on Lean Startup (Tauciuc, 2018)

Before talking about Lean startup, it is important to understand what Lean is. According to the Lean Enterprise Institute, Lean is a way to create more value for the customers using fewer resources. Lean started in the manufacturing field and the first concepts were devel-

oped by Henry Ford, when he wanted to speed the production of automobiles by integrating an entire production process via the assembly line in 1913. The concept brought production flow, but it couldn't provide variety in the automobile models produced. Toyota wanted to respond to the customers' need of variety and decided to revisit Ford's model by bringing new innovations and thus inventing the Toyota Production System. The main changes from Ford's model have been the switch of focus from the individual machines and their utilization to the flow of the whole process.

However, Lean evolved above being a manufacturing system and translated into a way of building software as well. It turned into a way of thinking that evolves around knowing what customer finds as being valuable, having a continuous flow of the production and critically assessing the steps involved in the process so that less resources, including time, are used (Womack & Jones, 1996). Lean has become a mindset – “a mental model of how the world works” (Poppendieck & Poppendieck, 2014). The lean mindset could be explained by a simple equation (Poppendieck & Poppendieck, 2014):

$$\text{Expertise} = \text{challenge} + \text{coaching} + \text{progress} + \text{perseverance}$$

Moving from manufacturing physical products to building software systems, the next natural step for Lean was to provide a framework for the entrepreneurship mindset. That is, not focusing only on the “building” phase, but also on the concept and creation phase. That is because, at least in the software world, these phases are intertwined with the changes in how the organizations are managed. The evolution of the management systems used in organizations have been showing a shift towards empowering more the employees (Laloux, 2014), and this lead to changing the mindset of employees from mere “working hands” to being part of the creative process. Ries calls an employee from a company creating a new product or service an intrapreneur.

Besides the already established organizations, there are the startups. A startup “is a human institution designed to create a new product or service under conditions of extreme uncertainty” (Ries, 2011). Someone building a startup is an entrepreneur.

Both entrepreneurs and intrapreneurs share the same mindset: the entrepreneurship mindset. By adapting lean thinking ideas to entrepreneurship, Ries created the Lean Startup concept (Ries, 2011).



He defines the main goal of a startup as discovering what is it that the customers want and are willing to pay for, and to do this discovery as quickly as possible. Lean Startup positions itself as a new way of tackling the development of innovative new products or services.

The 5 key principles of Lean Startup model are (Ries, 2011):

1. **Entrepreneurs are everywhere.** As discussed above, entrepreneurs are both startup builders and company employees, so Lean Startup can be applied with any size of company and at any stage of the company development.
2. **Entrepreneurship is management.** As the scope of a startup is not only to build a product or service, but to build something that the customers find valuable and are willing to pay for, it is not enough to make sure the product is built on time and on budget. There needs to be a good framework for listening and understanding the customer needs, such as product prioritisation decisions, constant testing and feedback. This means a new way of doing management.
3. **Validated learning.** The goal of startups is to learn how to build a sustainable business. That is why, all activities done in a startup (every product, every feature, every marketing campaign etc) are understood to be experiments designed to achieve validated learning. Validated learning is what also promotes the idea of failure as something desirable: if you cannot fail, you cannot learn. Some ways to achieve validated learning are: quick launch of the service, talking to customers, and continuously evaluating the efforts between creating value to the customer and creating waste. The concept of validated learning is crystallized further in the Lean UX philosophy. "**Lean UX** centres around validating hypotheses. [...] Instead of thinking of a product as a series of features to be built, Lean UX looks at a product as a set of hypotheses to be validated. In other words, we don't assume that we know what the user wants." (Klein, 2013).
4. **Build-Measure-Learn.** This is the core of the Lean Startup. Startups turn ideas into products, measure how customers respond, and then learn whether to pivot or persevere. The quickest build-measure-learn cycle that produces a valuable product or service for the customer is called a minimum viable product (MVP). The purpose of the MVP is to start the learning process, and offer the customer a competitive advantage, by taking the risk of launching something that the competition does not have yet – even though it is not 100% ready. When the service is launched, the customer can learn from the experiences of the users and go into a new cycle of build – measure – learn. Ries also calls this cycle the continuous innovation. (Ries, 2011).
5. **Innovation accounting.** This principle is tightly linked to "entrepreneurship as management". Since Lean Startup is favouring solving complex problems by innovative solutions, it also needs new paradigms for metrics: how to measure progress, set up milestones, and how to prioritize work.

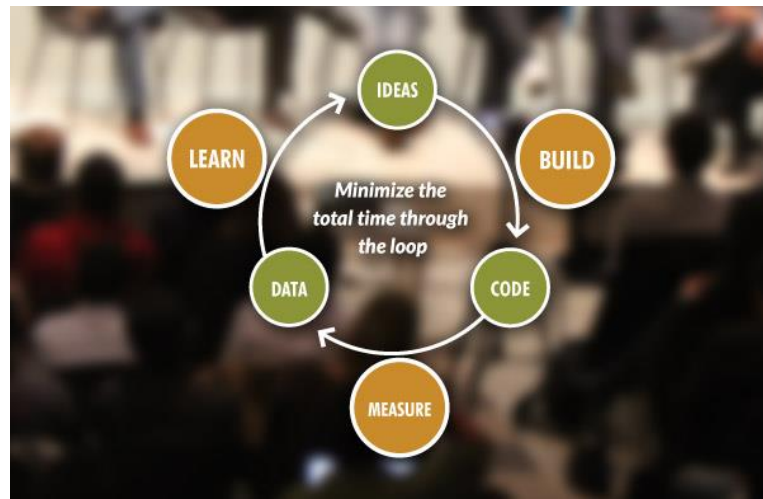


Figure 6. The Lean Startup process diagram (Ries, 2018)

### 2.3 Design thinking and service design

Design thinking is the 3<sup>rd</sup> and final building block on which LSC is based:



Figure 7. The research question, focus on Design Thinking (Tauciuc, 2018)

Unlike agile software development and lean startup, design thinking is a fairly recent concept, and there is no consensus as to what it exactly is (Nixon, 2016). Design thinking as a term has been first mentioned in 1987 by Peter Rowe, professor of architecture at the Harvard School of Design (Nixon, 2016).

The definitions range from very comprehensive ones like the one provided by the IDEO institute, to more open ones, like the ones provided by design studios. For example, IDEO's president and CEO, Tim Brown, defines it as:

"Design thinking is a human-centred approach to innovation that draws from the designer's toolkit to integrate the needs of people, the possibilities of technology, and the requirements for business success" (Our approach: Design thinking, ideo.com). In other words, design thinking is "a discipline that uses the designer's sensibility and methods to match people's needs with what is technologically feasible and what a viable business

strategy can convert into customer value and market opportunity” (Brown, as quoted by Martin, 2009).

Buchanan argues that “Frankly, one of the greatest strengths of design is that we have not settled on a single definition. Fields in which definition is now a settled matter tend to be lethargic, dying, or dead fields, where inquiry no longer provides challenges to what is accepted as truth.” (Buchanan, 2001) Having several evolving definitions, and a degree of uncertainty around the topic seems to contribute to the continuous innovation of design thinking.

Like lean, design thinking refers to “wicked problems” (non-linear problems that have a high degree of uncertainty). These problems are characterized by ambiguous or unclear boundaries to the problem and high uncertainty as to what might be the most successful solution. Many organizational problems are “wicked” in a similar manner (Nixon, 2016). To show what kind of process solving these problems required, Newman came up with the design squiggle:

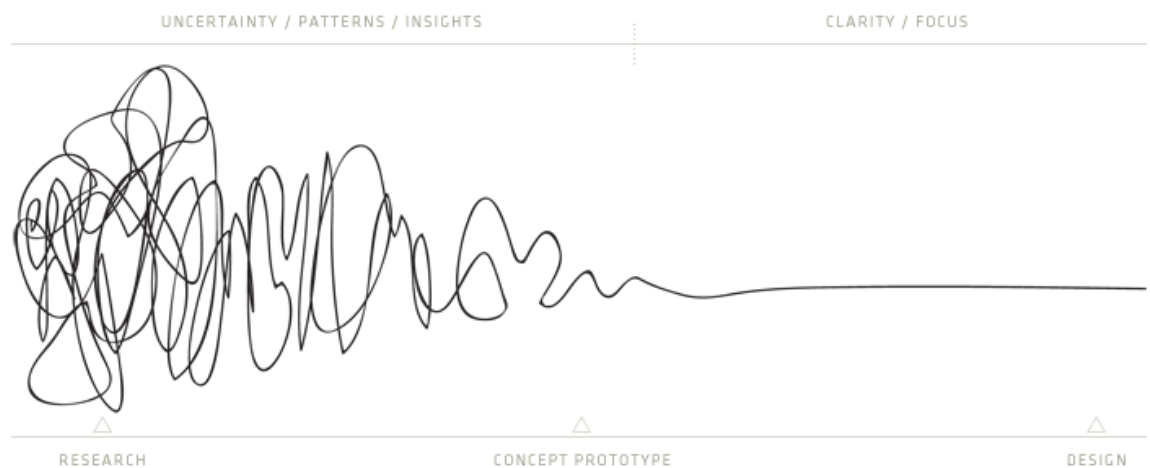


Figure 8. Solving a wicked problem (Newman, 2010)

To make the concept of design thinking more concrete, and to facilitate its applicability in the mainstream organizations, Liedtka & Ogilvie developed a design-thinking toolkit for organizations. The basis of the toolkit is the hypothesis of what would be the impact on the organizations and their activity if the managers thought more like designers. Their answer was that empathy, invention and iteration would be concepts ingrained in how the organizations operate and they envisioned a problem-solving process that has four stages (Liedtka & Ogilvie, 2011):

1. *What is* – exploring the current reality
2. *What if* – envisioning a new future
3. *What wows* – making choices from the possible future scenarios
4. *What works* – validating these choices against the marketplace

Each phase of the problem-solving process has different tools mapped to it, as shown in the figure below:

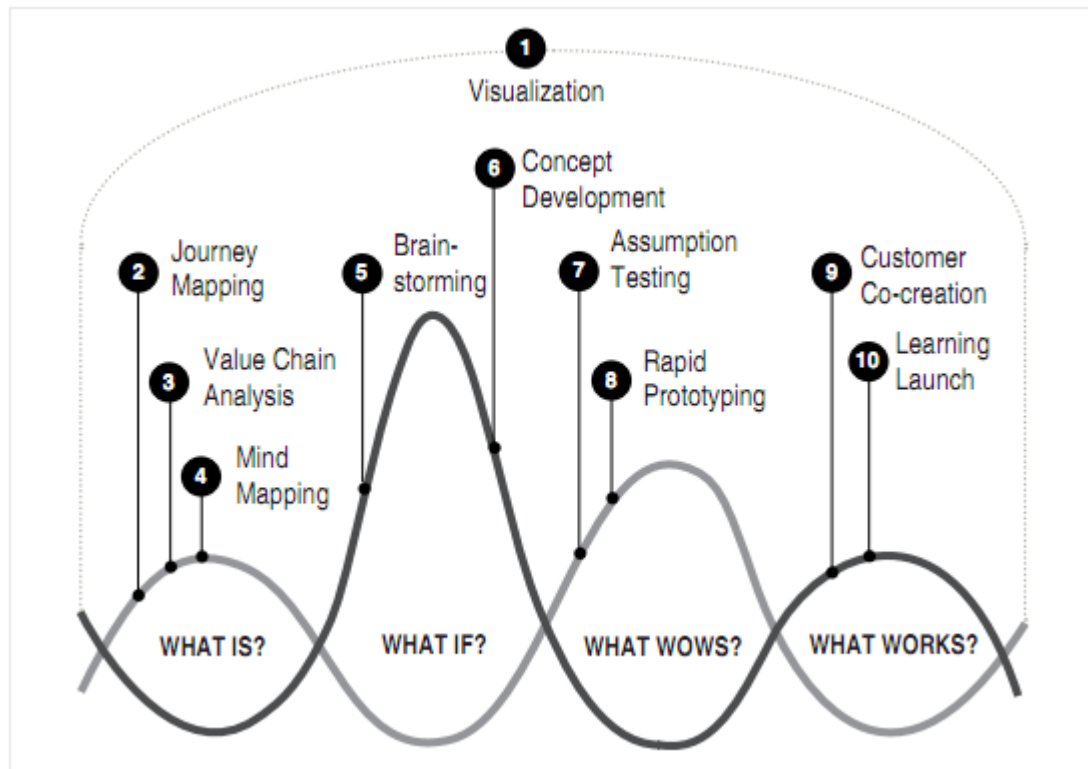


Figure 9. Visualization of the design thinking process (Liedtka & Ogilvie, 2011)

Thus, design thinking focuses not on finding the one true solution (which is impossible for such complex problems where several stakeholders are involved), but rather in modelling potential scenarios together with the potential users/beneficiaries of the solution for the problem. These scenarios are then conceptualized as prototypes to be validated with the users.

Ingle defines further this process into five phases of design thinking, which are rather similar to Liedtka's and Ogilvie's approach (Ingle, 2013) and also to the process the Institute of Design (D. School) at Stanford defined:

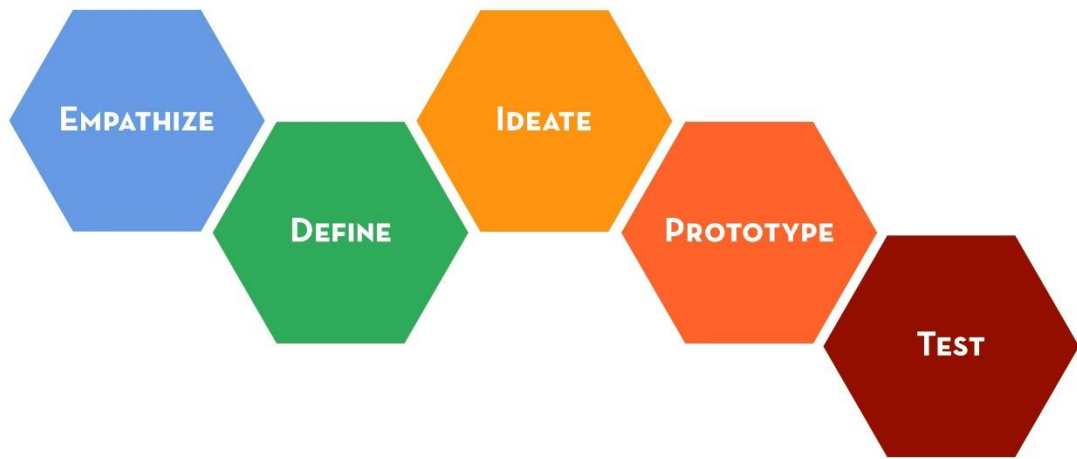


Figure 10. Design thinking process (Institute of Design at Stanford, 2012)

Let's look at the phases in more detail:

1. **Phase 1**, also known as empathy. Understand the business challenge, beyond previous experience with similar challenges. The stepping stone is that in understanding the current challenge, the problem solver does not simply rely on historical data, but rather has an active dialogue with the potential users. The main activity during this phase is the research, which should be purposeful, affordable and actionable, to uncover the actual needs that the users have and may not be able to express directly. A tool used often in research is “ask why”, which focuses on asking why as many times as needed to get to the actual needs. Sinek is also arguing that the most important thing to understand someone is not to know what they do or how they do it, but rather what motivates them – their why (Sinek, 2011).
2. **Phase 2**, the define phase. During this phase, the focus is on defining and refining the challenges to be solved, using the information gathered during the empathy phase. The reason why a challenge needs to be solved should also be uncovered, in order to get a full understanding on the challenge to be solved.
3. **Phase 3**, the ideation phase. This is the moment to be creative and imagine different solutions and future scenarios for the defined challenge. Sometimes service/product creators are tempted to jump the creation process straight to ideating services and features. This tendency brings the situation when the solution is not actually fixing the correct problem, so the product or service ends up not being utilized by any users. When it comes to ideation, there are different techniques used, from which the most widely known may be brainstorming. The result of the ideation phase is a series of concepts that need to be experimented with the users to validate their marketplace suitability.
4. **Phase 4**, prototyping. The concepts from the ideation phase are usually intangible (e.g. services, experiences, processes, some products). Translating these concepts into tangible items plays a crucial role in facilitating their validation. Prototyping is the process of transforming intangible concepts into tangible ones. Prototypes are representations of the concepts that help bring the ideas forward to others (including users).

5. **Phase 5**, testing. During testing, the service creators are going back to the users they empathized with and test the prototypes with them. The purpose of the testing phase is to validate the ideas with the target user group. As design thinking is entwined with the lean mindset, testing needs to happen as early as possible, so that cost is saved in implementation. The learnings from the testing phase are used to modify the concepts behind the prototypes, and a new build – measure – learn cycle starts. The concepts aim to be in the sweet spot between doable ideas, resources and impact.

The process of design thinking, as its definition, is visualised slightly different by different entities, but all the processes have at core the build-measure-learn concept:

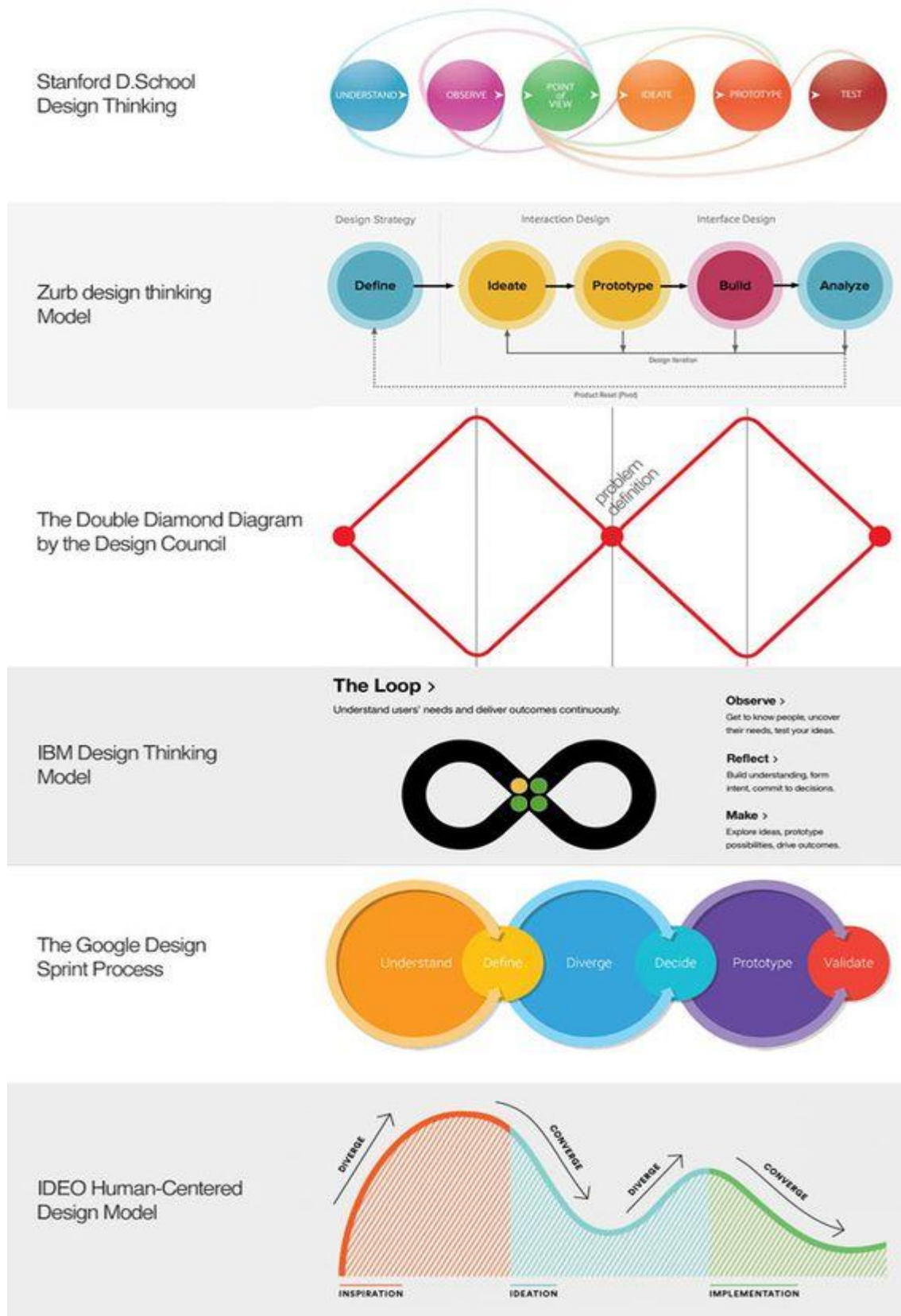


Figure 11. Distinctive design thinking models (<http://www.designorate.com>, 2018)

In the context of this study, it is interesting to understand how design thinking may help solving problems in organizations. Nixon mentions three main reasons (Nixon, 2016):

1. The complexity of organizations. As organizations are getting more and more complex (see chapter 2.6 New generation of organizations), they face problems that are novel and unique compared to the problems organizations faced tens of years ago. In this situation, planning cannot help, as planning is based on having prior experience with the problems. The situations are high uncertainty and require innovative solutions – just what design thinking is designed to deliver.
2. Decision makers' limitation to always find new and ground-breaking ideas. As humans, decision makers tend to project their own thoughts and preferences, and find justifications for the solution they already have, rather than look at the problem more closely. Design thinking is focusing on the user, thus switching the spotlight from the perfect solution to the correct problem.
3. Difficulty to take risks in stressful situations. In certain situations (e.g. responding to environment changes, including competitors), the fear of failure may overshadow taking necessary risks. Design thinking, with its experiments approach, helps diminish the feeling of failure and rather focus on the learnings and reiterating.

Solving a problem in an organization means, essentially, solving a challenge in a service that the organization offers its employees. Thus, organizations need to design these services, and design thinking, as we've seen, is a helpful tool for service design.

At a broader level, service design is an interdisciplinary approach that combines different methods and tools from various disciplines to solve complex problems (Stickdorn & Schneider, 2012). As it is an emerging and live approach, like design thinking, service design has a wide variety of definitions, ranging academic angles to design agency definitions.

For example, The Copenhagen Institute of Interaction Design defines service design as “an emerging field focused on the creation of well thought through experiences using a combination of intangible and tangible mediums. It provides numerous benefits to the end user experience when applied to sectors such as retail, banking, transportation, and healthcare. Service design as a practice generally results in the design of systems and processes aimed at providing a holistic service to the end user. This cross-disciplinary practice combines numerous skills in design, management and process engineering. Services have existed and have been organised in various forms since time immemorial. However, consciously designed services that incorporate new business models are empathetic to user needs and attempt to create new socio-economic value in society. Service design is essential in a knowledge driven economy.” (The Copenhagen Institute of Interaction Design, 2008)

Meanwhile, the design agency Volts Service Design has the following definition: “When you have two coffee shops right next to each other, and each sells the exact same coffee



at the exact same price, service design is what makes you walk into one and not the other.” (Volts Service Design, 2008, as cited by Stickdorn & Schneider, 2012).

The service design way of thinking, which can be applied to any kind of services (internal or external), has five principles (Stickdorn & Schneider, 2012), curated graphically below:

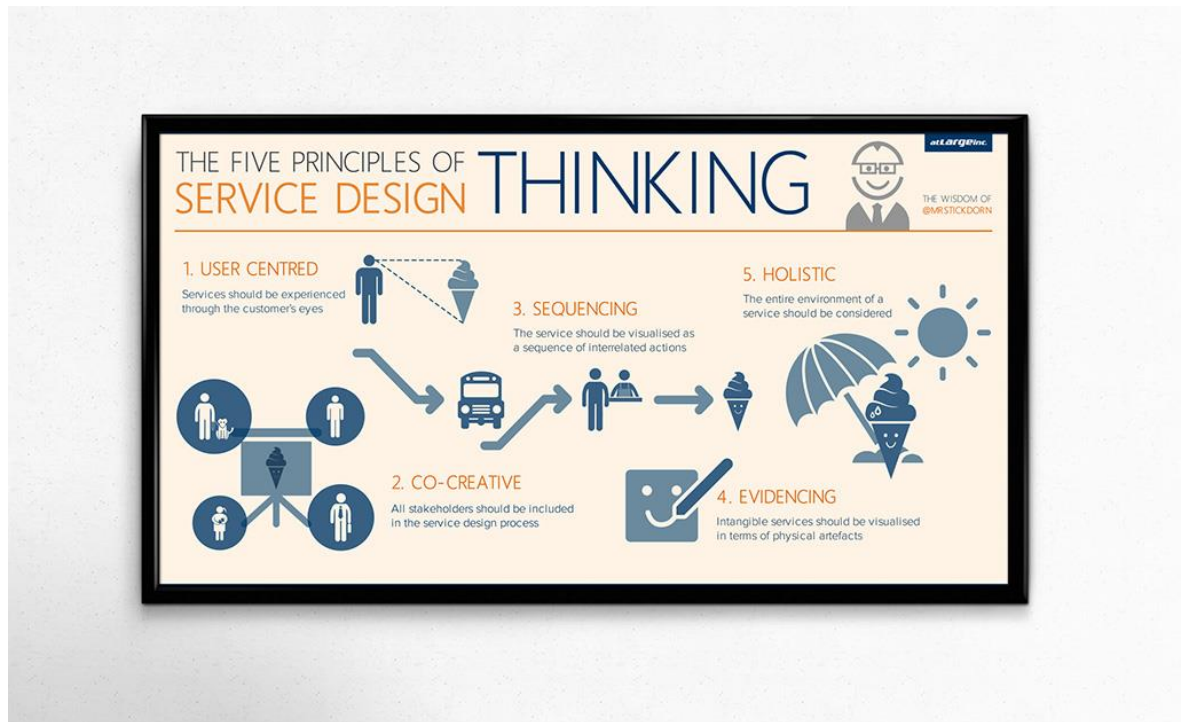


Figure 12. The five principles of service design thinking (Tyass, 2014)

Service design thinking has a strong influence on LSC, as we will see in the next chapter.

## 2.4 Lean Service Creation

Now that the building blocks of Lean Service Creation (LSC) have been discussed, let's take a closer look at how they influence LSC and how LSC is different.



Figure 13. The research question, focus on LSC (Tauciuc, 2018)

LSC inherits the build-measure-learn from the lean startup, the experimentation and user centricity from design thinking and the agile development approach. However, it builds further on top of these concepts to focus on team work and co-creation.

LSC brings the main tools from the three different schools of thought to one single process, so the team can have a common language, regardless of the background of the team members, and can tackle the whole creation process. Nevanlinna argues that none of the three schools of thoughts tackle all the main points to be considered when creating a new service (Nevanlinna, 2018):

- Agile has boxed itself as a software development process. It is not about creating the business or service, so it is directed to the tech skilled people.
- Design Thinking is way too much about the user centricity, way too often forgetting that creating a successful service is tightly related to serving the business needs.
- Lean startup gives a good framework for iterating a concept but does not tackle finding the user needs or building the service.

According to the LSC manifesto, LSC “is a systematic and adjustable way for multidisciplinary teams to create new services.” (<https://leanservicecreation.com/manifesto>). At a high level, LSC is made of:



Figure 14. LSC components (Futurice, 2017)

LSC started with an initiative to change how the company was selling and doing projects. Drawing on its building blocks, the LSC mindset is built around human centricity regarding the team work, customer centricity, transparency, continuous learning, high-level and detail-level focus. Employees with design, business and tech backgrounds worked together as a team to sketch the main phases of the process: finding the problem worth solving, finding the product market fit, and growth hacking. The general LSC process has been defined as:



Figure 15. LSC service creation process. (Futurice, 2013)

The next step was when one of Futurice's customers asked Futurice if we can teach them the way we use to create digital services. At this point, the first set of tools was put together. It contained lectures and workshops around lean startup, design thinking and agile philosophy. This defined the practices and way of working.

During the next iteration, the first canvases were created. Their purpose was to define the process even further and make it easy to follow. (Nevanlinna, 2018) The canvases are tools to ask the right questions during the process, and not to document. They bring more tangibility and order to the process and are constantly evolving. Below are the canvases from the LSC Fullstack version to be released in 2018:

## LEAN SERVICE CREATION CANVASS BREAKDOWN

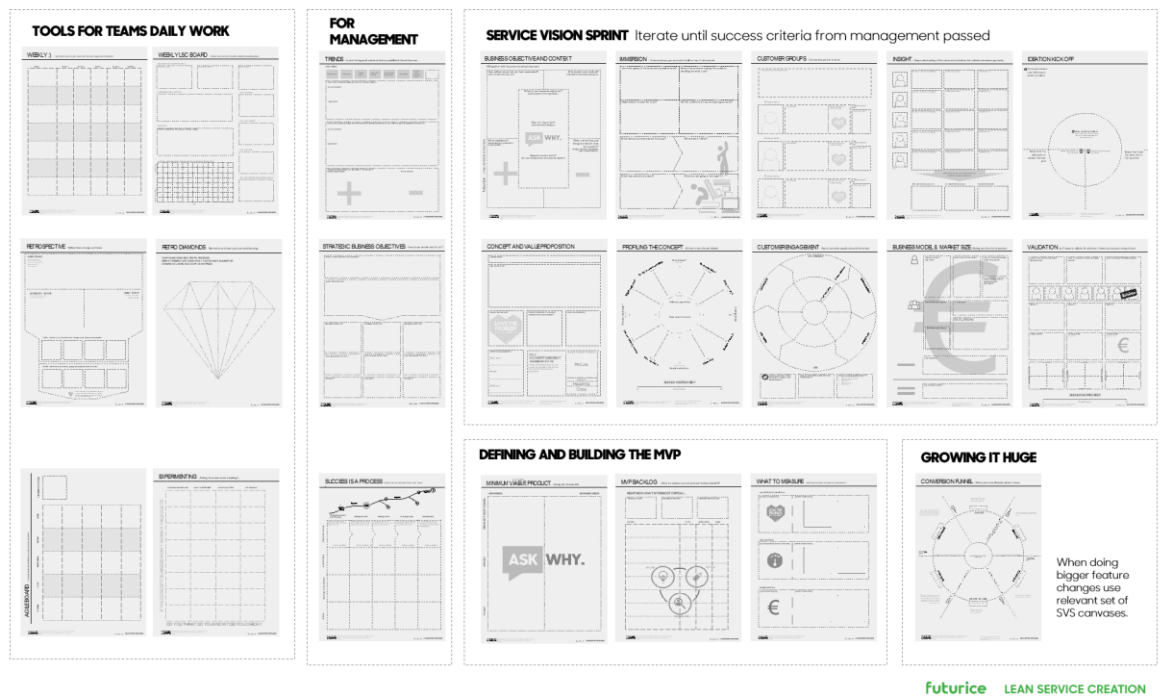


Figure 16. LSC Fullstack set of canvases (Futurice, 2018)

As said, the purpose of the canvases is to make the process of service creation more tangible and straightforward. Even though the set is extensive, each team can and should choose the canvases appropriate for their own work, that they find useful.

LSC is a concrete process, straightforward (easy to understand and teach, does not need extensive training to be applied, having the canvases is enough). It has a rather strict workflow that creates trust, helps the team focus on the right topics and it is easy enough to take into use in different type of organizations and challenges.

According to the manifesto, LSC “has a social mission: to make the best practices of design & development freely accessible to everyone. That is why LSC is free to use, free to adapt, and free to grow. This has made LSC probably the most validated and used open source service creation process on the planet.” (Futurice, 2017)

The approach of combining agile, lean startup and design thinking is spreading, as can be seen from this graphic published in 2016 by Gartner:

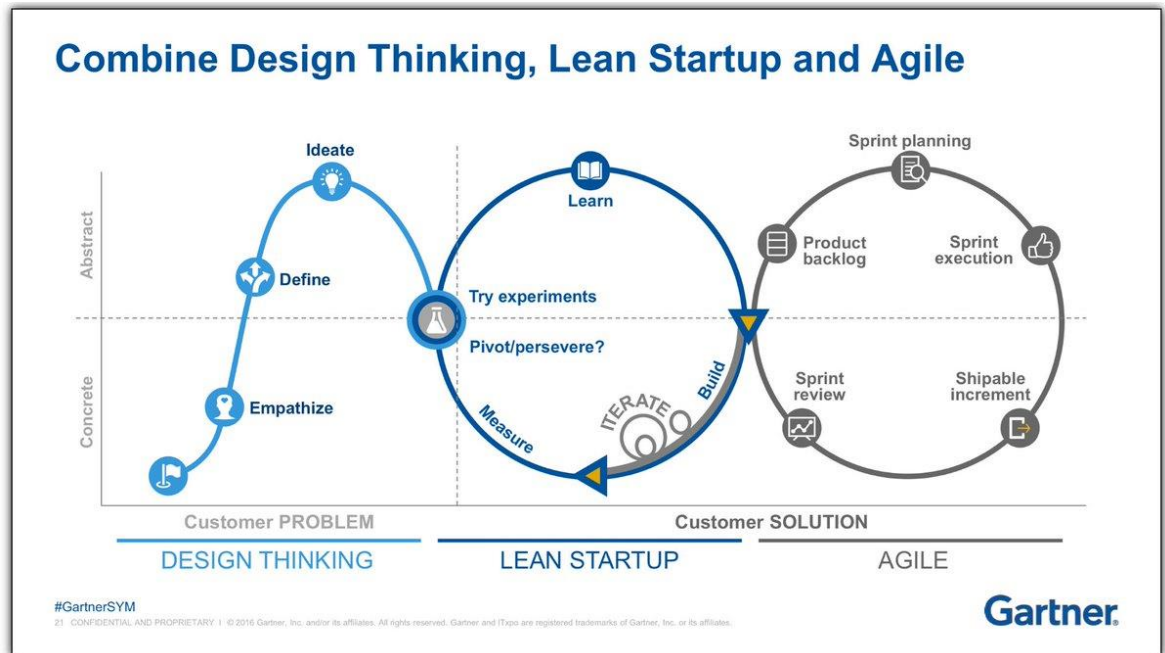


Figure 17. Gartner approach on combining lean startup, agile and design thinking. (Gartner, 2016)

## 2.5 Continuous software development and software maintenance

The next item in the research question is the software maintenance or continuous software development:

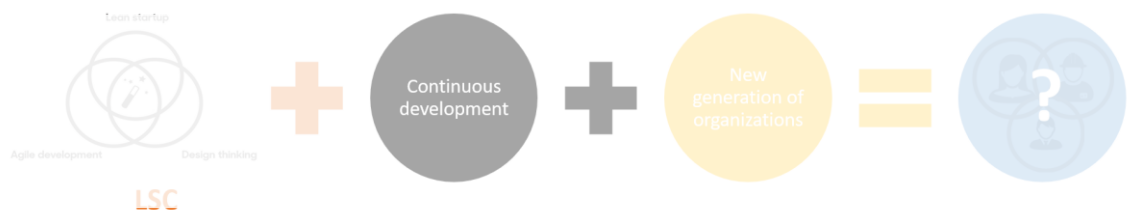


Figure 18. The research question, focus on continuous development. (Tauciuc, 2018)

While LSC, with its building blocks, refer mostly to creating new (digital) services, and bring new approaches to tackling the beginning of a service lifecycle, there seems to be more struggle in the process once the MVP is launched.

In a traditional waterfall approach, we talk about launching a ready product. What comes after the launch is the software maintenance. Software maintenance is “the act of keeping

an entity in an existing state of repair, efficiency, or validity; to preserve from failure or decline.” (Grubb & Takang, 2003). It goes forward to say that the software maintenance is a “modification of a software product after delivery, to correct faults, to improve performance or other attributes, or to adapt the product to a modified environment” (Grubb & Takang, 2003). The software maintenance conscious lifecycle promotes the idea that maintenance exists only in the context of waterfall development:

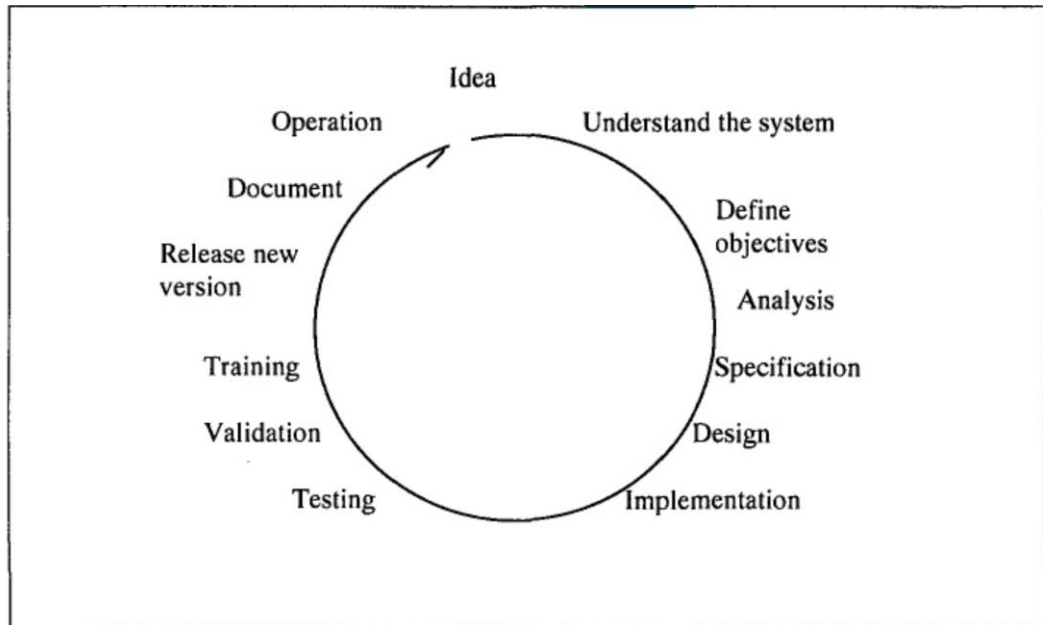


Figure 19. The "maintenance conscious" lifecycle. (Grubb & Takang, 2003)

There are three views on the software maintenance (McDermid, 1991):

1. The bug- fixing view – maintenance is seen as the detection and corrections of errors in already launched services.
2. The need-to-adapt view – when the operational environment and original requirements of the service change, maintenance is adjusting adapt to these changes
3. The user-support view – maintenance is offering support to the users

The three views grant software maintenance the role to fulfill the following needs that a launched service has (Grubb & Takang, 2003):

1. Provide continuity of the service – once launched, a service needs to be reliable. Today, when more aspects of the society are relying on digital services, the continuity of the services is critical. This aspect is ensured by having mechanisms such as issue fixing, recovering from failure, and accommodating changes in software and hardware. In LSC words, with launching a service, the organization makes a promise to their users. This promise can be kept by launching a viable product and enriching it with viable experiments.

2. Support mandatory upgrades – these changes may come from changes in legislation or from the need to keep a competitive edge over rival products. LSC treats these upgrades as iterations following the “learn” phase. Learning can come from inside the product, from external forces or from observing the competitive market.
3. Support user requests for improvements – the more relevant a system is to its users, the more users will make use of the system. This concept is integrated into LSC, as its scope is to create lovable services, that users delight in using. A special focus is given to user feedback, as the usage information is the key to keeping the service relevant. “Data is the king when the service is launched.” (Nevanlinna, 2018)
4. Facilitate future maintenance work – essentially, it refers to creating quality code and remembering that shortcuts taken during software development are costly in the long run. In the context of iterations, the current iteration is the basis for the next iteration. Having a sturdy base makes the development faster and more agile.

Software maintenance is, in fact, the creation, development and validation of new iterations of an already built service. However, in the context of today’s innovative concepts, user centricity and speed of delivery, talking about software maintenance brings a defensive attitude. Even most of forward thinking companies like Futurice’s customers, who believe in the build-measure-learn cycle, still see the MVP as a finished product (Nevanlinna, 2018) and everything that comes after the launch as maintenance. Maintenance is seen as a cost, and this brings a slowing down of the build-measure-learn cycle and some services may become irrelevant to their users.

This means that the software maintenance has an image problem and it needs rebranding. It is time to update the vocabulary used for the post-launch phase of service, just as we updated the vocabulary for the pre-launch phase, because “traditional lifecycle models fail to take account of the evolutionary nature of software systems” (Grubb & Takang, 2003). We are not using traditional software development models anymore, but models that recognize that digital services are living entities. We could stop talking about “maintenance”, but rather help the companies perceive it for what it is: “continuous software development”, or the post-MVP launch phase covered by LSC.



Figure 20. LSC process, focus on post-launch of MVP. (Tauciuc, 2018)

## 2.6 New generation of organizations

The study has been conducted in Futurice, which is a non-traditional organization creating digital services. Due to this context, the last piece of the research question refers to a new generation of organizations:



Figure 21. The research question, focus on new generation of organizations. (Tauciuc, 2018)

Why do we talk about a new generation? In the past 25 years, the way software is being built has changed dramatically, from a waterfall model to a more innovative model, based on user and customer centricity and the concept of continuous feedback loop. These new methods of engaging the market bring new leadership approaches. (Gothelf & Seiden, 2017).

Laloux offers a comprehensible view on the evolution of the organizations, that comes from the evolution of the leadership approaches:



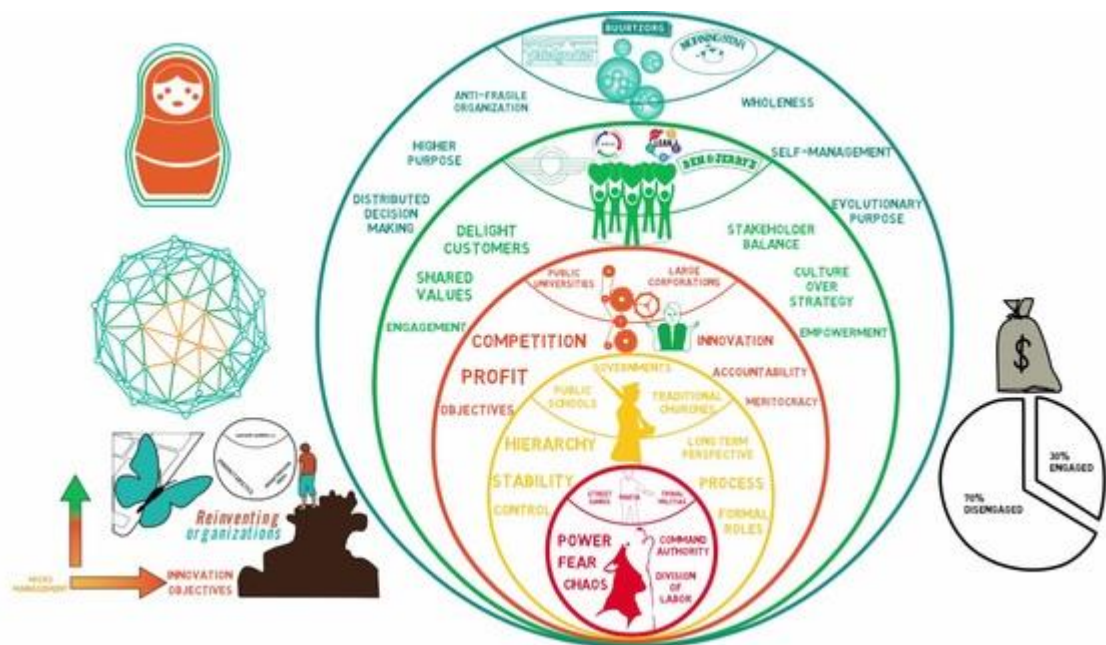


Figure 22. Main organizational paradigms according to Laloux. (Collins, 2016)

Organizations have evolved from tribal structures (Red) to more formalized hierarchical structures (Amber / Orange), to agile approaches (Green / Teal):



Figure 23. Main characteristics of organizational paradigms according to Laloux.

(<https://medium.com>, 2016)

The predominant frame of reference in management thinking today is the orange organization. This kind of organization can be conceptualized as a machine that uses innovation as the key tool to beat its competition, achieve profit and growth. The key breakthroughs that this organization type brings are: innovation, accountability and meritocracy. The management model is based on objective: control the *what*, freedom on the *how*. (Laloux, 2014).

The philosophy of controlling the content (the *what*) and giving freedom on the process (the *how*) is contrary to Hock's view. Hock, the founder and former CEO of VISA, believes in chaordic organizations and has proved the effectiveness of the concept by modelling VISA as a chaordic organization and leading it to its success (Hock, 2005). He coined the term "chaordic" with the following definition:

***chaordic*** \ kay'ord-ick \ adj. [fr. E. *cha'os* and *ord'er*] 1. The behavior of any self-organizing and self-governing organism, organization, or system that harmoniously blends characteristics of chaos and order. 2. Characteristic of the fundamental, organizing principle of nature

Hock argues that the more a process is defined, the more it gives room for the content to be chaotic. In other words, makes more room for innovation.

The chaordic concept is more in line with the green and teal organizations. The green organization model is the next model that appeared after the orange organization. As opposed to the orange – machine, the green organization's metaphor is the family. The green organizations are typically culture driven, and they bring important innovations such as empowerment, values-driven culture and stakeholder model. They do not focus on beating the competition anymore, but on achieving extraordinary employee motivation (Laloux, 2014).

Building on top of green organizations, Laloux argues that the next level, which is developing today, is the TEAL organization. This model brings three breakthroughs (Laloux, 2014):

1. Self-management – the system is based on peer relationship, without the need for hierarchy. One example of this could be the 3x2 model used in Futurice for decision making.
2. Wholeness – the organization welcomes the employees as whole beings, with all their aspects. They do not expect to see only the professional self or determination, but the emotional and intuitive parts of the employee are also welcome. This creates a safe space where creativity can thrive.

3. Evolutionary purpose – teal organizations are perceived as having a life and sense of direction of their own. Instead of trying to predict the future, members of organizations are invited to listen in and understand what the organization wants to become. This principle has been applied in Futurice when the vision 2023 was defined and co-created with the whole company, not only by a group of chosen few.

These green/teal organizational models are more than a process, they are a new culture. Seiden calls this culture “sense & respond”. (Gothelf & Seiden, 2017) In a nutshell, the concept of sense & respond can be visualized such as:



Figure 24. Sense&Respond metaphor. (Voinonen, 2015)

Sense & respond is governed by five principles (Gothelf & Seiden, 2017), that embody also what LSC stands for:

1. Create two-way conversations: the focus is on listening to the market, making a credible guess, getting feedback almost in real time, and adjusting. LSC calls this “love the problem, not the solution”: focus on understanding and defining the right problem and iterate through solutions.
2. Focus on the outcomes: stop focusing on output (features), but rather on outcomes (business outcomes management wishes to achieve and find a way to fulfil them). In practice, it is about creating the conditions for teams to try different approaches, experiment, learn, discover through trial and error. LSC process starts with the business need, which the whole service design is based on.
3. Embrace continuous change and continuous processes. The build-measure-learn cycle is the core of LSC.

4. Create collaboration: consider the way the teams are organized. LSC's starting point is to create a common language that facilitates collaboration in the team.
5. Create a learning culture: requires openness, humility, and permission to fail, while supporting curiosity and collaboration. LSC's manifesto mentions to "never stop iterating, never stop learning" (<https://leanservicecreation.com/manifesto>)

It is important to note that organizations may have different levels of maturity in different departments or even teams. However, LSC can exist only in an environment which is culturally compatible with the LSC values and concepts. This environment embodies green/teal organization principles, or a sense&respond culture, which can be combined with a chaordic approach that boosts creativity.

This chapter concludes the building blocks of the research question. As shown, all blocks are interrelated.

## 2.7 Premises for the study

The original question of the study is:

How can LSC tools help developers overcome their problems during the software continuous development phase?



Figure 25. The research question (Tauciuc, 2018)

The previous chapters explored how each component of the question is translated in more tangible terms, and we discovered that:

- LSC is a way of creating new services based on user centricity, customer centricity and continuous learning. It aims to offer a common language for interdisciplinary – multiskilled teams and facilitates their work with a straightforward process supported by an easy to use set of tools. LSC tackled the whole lifecycle of creating a service, starting with the business needs, continuing with the MVP and then further on with continuously improving it. The model is currently more developed for the beginning of the lifecycle.

- Continuous software development is the natural step that comes after the MVP. It can encompass anything from fixing bugs to adding new major features to the service. Before known as software maintenance, the concept is perceived as being old-fashioned and a cost for the service. However, the vocabulary around software maintenance needs to change towards continuous software development, to facilitate keeping up with the new software development models (e.g. agile).
- The new generation of organizations are self-organizing and adaptive, based on a sense&respond culture that incorporates a chaordic approach. The way services are built has changed, the way the teams are built and the way they interact has changed, and this triggers new leadership and organizational models.

Since the three concepts have similar values, the purpose of the study is how to promote the change from software maintenance towards software continuous development in the daily work of the team doing the continuous development, after the MVP is launched.

One way to tackle the study is to focus on the following three areas. The same approach is also used when introducing an agile approach to a project (PMI, 2017):

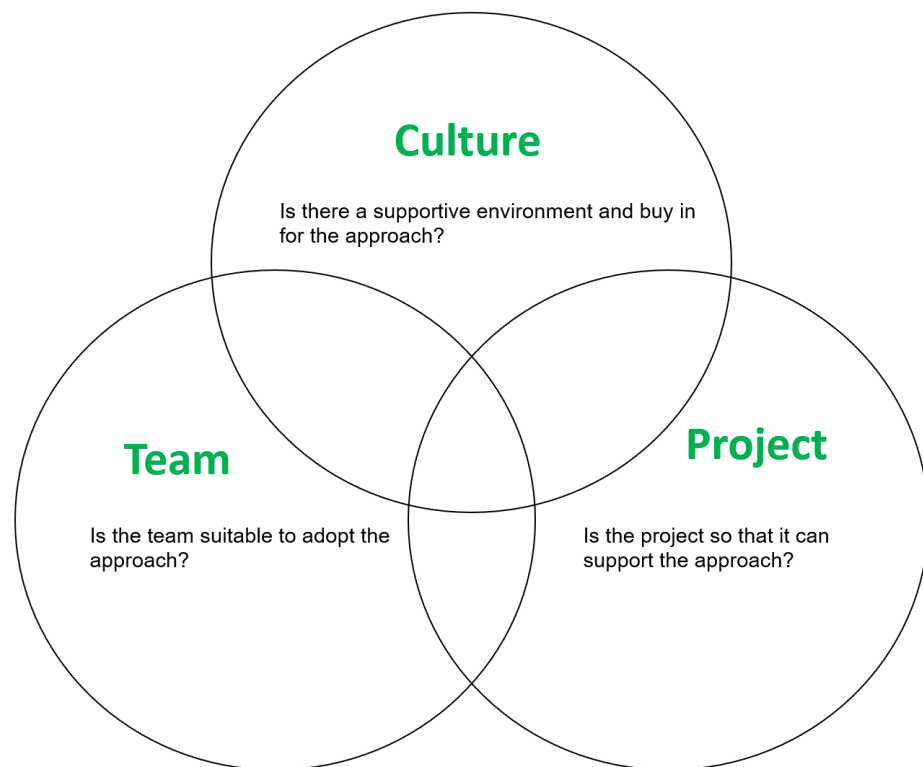


Figure 26. The three areas that help solving the research question. (Tauciuc, 2018)

### 2.7.1 The Culture: is there buy in?

As seen in the chapters above, the culture of the environment (Futurice) is ripe for extending LSC in the post – MVP launch phase. LSC has been used in creating new services, as well as in growth hacking existing services (Nevanlinna, 2018).

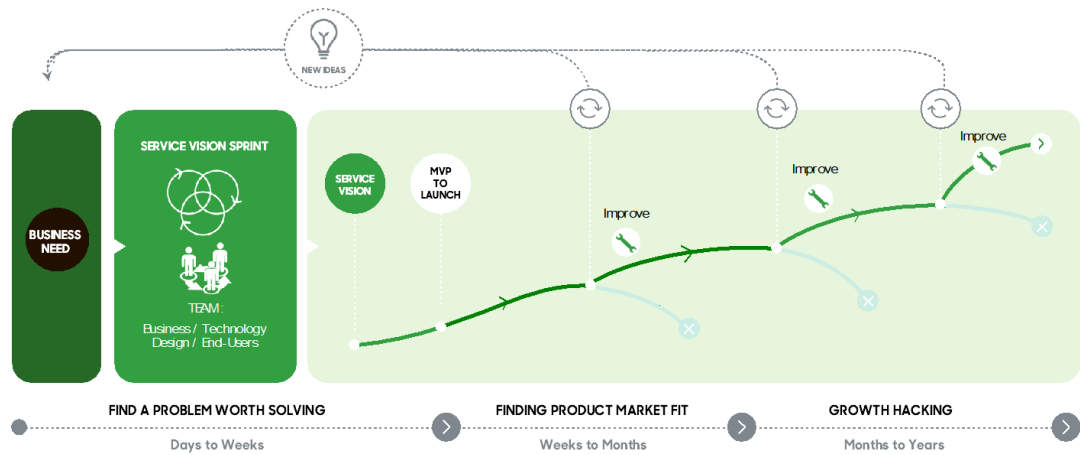


Figure 27. The LSC model. (Futurice, 2013)

From the figure above, we can conclude that LSC is already tackling the continuous development from the mindset and process perspective.

However, the toolset is not thoroughly extended yet, especially for the cases when bigger changes need to be implemented, as they require an open conversation with the customer. This conversation would benefit from taking more canvases into use, as there is a clear business goal (Nevanlinna, 2018).

The canvases would also open the discussion with the customers about what a MVP involves, and why is there a need for continuous development. Currently, customers are used to buying MVPs – they believe that when it's done, it's done and ready (Nevanlinna, 2018).

### 2.7.2 The Team: is it suitable and ready for LSC?

LSC is a customer centric mindset, and its aim is to bring value to the customers. A customer value-oriented service is built on engaged customers and engaged employees (Moreira, 2017).

As the team is part of Futurice, it needs to function in a new organization setting with a sense & respond culture. This means that the team needs to be small, cross-skilled, autonomous, experimenting and learning in pursuit of a vision or a strategy. This team needs a greater decision-making authority and is in constant conversation with the customer (Gothelf & Seiden, 2017).

A team working in a customer-value-driven (CVD) organization has the following traits (Moreira, 2017):

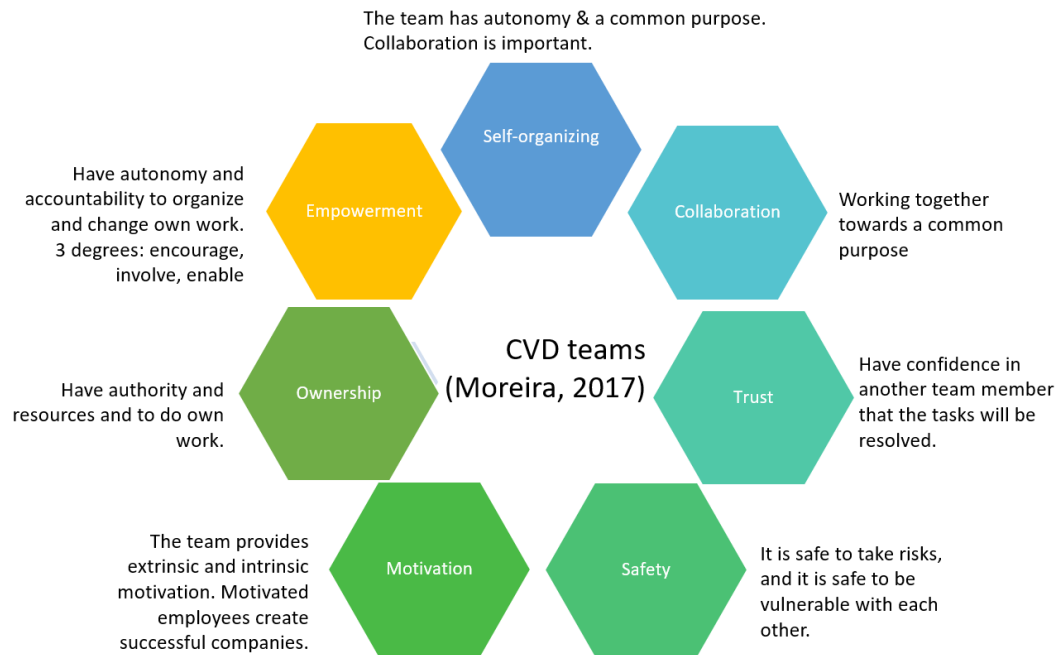


Figure 28. Characteristics of customer-value-driven teams, according to Moreira, 2017. (Tauciuc, 2018)

Leffingwell brings also interesting insights on the high performing team (Leffingwell, 2007):



Figure 29. Traits of high performing teams according to Leffingwell, 2007. (Tauciuc, 2018)

Also, the team needs to work in a scrum environment. Essentially, in looking at how suitable the team is for the new approach we need to look at how the team applies the principles mentioned above, including the scrum values:



Figure 30. Overview of traits for CVD teams, high performing teams and scrum teams.  
(Tauciuc, 2018)

All the values above are in line with the values promoted by LSC. As the team is already embodying several of the building blocks above, the team is definitely suitable and ready for extending the LSC approach.

### 2.7.3 The Work: does it permit LSC?

LSC has been labeled as a tool to create new services. However, it is strongly related (and has also been applied to) improving or growing existing services.

The key concept is to understand that the work is not only about the actual product or digital service that the team is building continuously, but also about all the other blocks around it: the customer service, the roadmap envisioning and planning, the Futurice brand and promise etc. There are two levels to the work:

- **The daily level**, when the team is handling refactoring of the code, fixing issues, planning and implementing new features, making the service more robust, etc.
- **The higher level**, that is customer-value-driven, that helps build strong relationships with the customers and makes the customers recommend the daily work. In other words, the service design.

The study will investigate applying the research question at both levels.



## **3 Research design**

### **3.1 Study setup**

In this chapter, we will look at the target and objective of the study, redefine and bring focus to the research question taking into consideration the theoretical framework discussed before, and present the general structure of the study, including the team setup and the author's role.

The purpose of this chapter is to present a solid foundation unto which the study is built.

#### **3.1.1 Target and objective of the study**

The target of this study is to use a LSC approach to find and test actionable items that could alleviate the main problems that the continuous development team is currently facing. This means that the focus will be on the build – measure – learn cycle: not find perfect solutions, but rather offer the possibility to use continuous learning.

The objective of the study is to look at the continuous development team's work as a service, not as a concatenation of daily tasks performed by parallel teams.

Once the main problem areas are identified, a list of suggestions (experiments) will be compiled. The experiments will be based on the theoretical framework and on ethnographic research of the team. For the duration of the study, the experiments will be run, and their impact will be assessed. Also, their belonging to the LSC toolbox will be discussed.

The objective of the study is not necessarily to question the applicability of all existing tools of LSC or to compile a new set of tools for LSC, but rather to have an impact in the current team's work and bring awareness to the post-MVP launch phase of LSC. Starting from one of the values of LSC ("love the problem, not the solution") the study aims to look at the real problems and find ways to alleviate them, rather than trying to see how existing solutions (existing LSC tools) could potentially alleviate the problems.

While there are many areas that can be tackled, the study will focus on the continuous development team itself. The focus is set so that it fosters the creation of actionable items and visible results. The experiments will target the team, their daily challenges and the service level challenges.

### 3.1.2 The continuous software development team's setup

From the organizational point of view, the continuous development team is a separate virtual team inside Futurice, with profit and loss responsibility. Its members are part of the official departments in the company, so they take part in the company's activities in those groups as well and are in touch with the other employees.

The team consists of mostly software developers and business and sales-oriented people. Depending on the customer case, people outside the team are invited for collaboration, for e.g. designers, data scientists, business advisors etc.

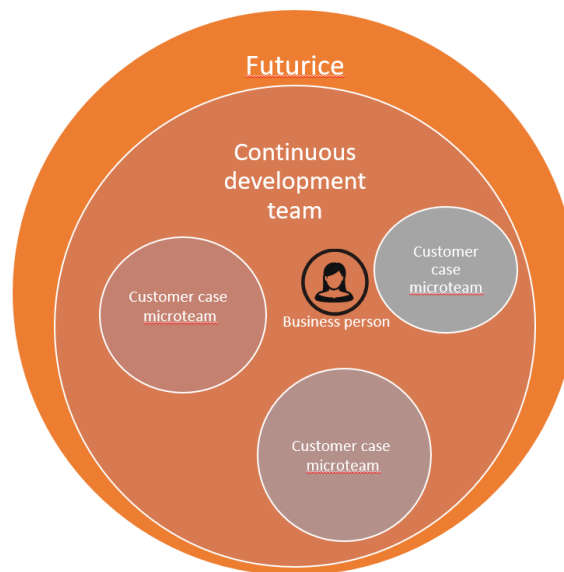


Figure 31. The continuous development team ecosystem. (Tauciuc, 2018)

The typical main differences between this team and any other project team that is creating a new service are the following:

- Each person in the team is dividing her work between 2-3 customer cases
- No one works alone: each customer case has a team, regardless on the amount of work to be done. These case teams are usually 2-4 people with interdisciplinary skills
- Each case team is in direct contact with the customer. They are organizing their work and being accountable for it
- The customer cases typically take several months or even years
- Each customer case has a specific bandwidth per month and the work is designed and agreed upon taking also this limit into consideration
- Each member of the team is encouraged to learn new technologies, and this is facilitated in how the work is organized
- Inside each case team, the team members regularly learn and help each other

Each case team has a service manager, that is involved in keeping the service level and strengthening the relationship with the customer. The service manager is generally outside the case team – not involved in daily work.

Looking at the overall team, there are other leadership roles involved: the technical leader, who provides guidance in technical matters and the business leader, responsible of the team's vision and strategy, profit and loss, marketing, business decisions.

### **3.1.3 The author's role in the study**

The author of this study has played different roles during the design and implementation of the work:

1. Lead of the Continuous Development unit in Futurice – in this role, the author has been responsible of the vision for the unit, development of the offering towards customers and the management and growth of the team. Also, the author had had overall responsibility on the profitability of all the customer cases.
2. Service Manager in customer cases – in this role, the author has taken part in 3 different customer cases during the duration of the study. As service manager, the author has been responsible of the overall customer communication and project manager-like activities.
3. Researcher – as researcher, the author designed and executed the phases of the study.

The design of the study, data collection, data analysis and results interpretation are done considering the three different angles that these three roles facilitated.

### **3.1.4 Defining the focus of the research question: what are the main problem areas to be addressed?**

The author has a wide view on the team and the different environments where the team was acting, due to having three roles for the duration of the study, as lead of the continuous development service and team, service manager in customer cases and researcher. To maximize the benefits coming from the multiple roles, the author of the study used ethnographic research to understand what main problems the team is facing. For this scope, it is useful to visualize the stakeholder map of the continuous development team. A stakeholder map “helps to visually consolidate and communicate the key constituents of a design project, setting the stage for user-centred research and design development” (Martin & Hanington, 2012). The continuous development team interacts with several stakeholders:

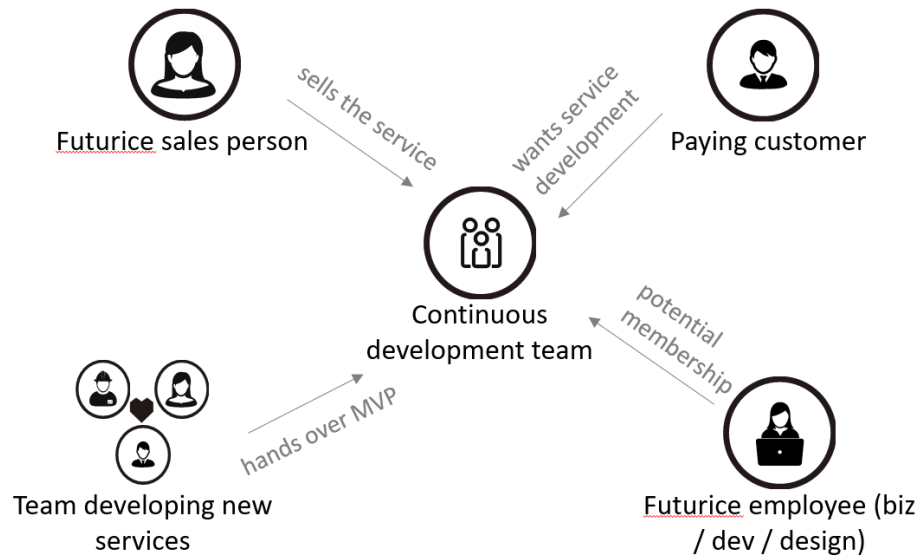


Figure 32. The stakeholder map of the continuous development team. (Tauciuc, 2018)

The team provides services to several types of customers:

1. **the customers who are paying for the services and their upkeep and growth.** Customers were expecting the same level of service, and the same attitude that they encountered during the MVP building. In this context, from the team's point of view, the key issues have been mostly around the customer not knowing exactly what to expect from the service as deliverables, the perception of continuous development as a cost and not enough information on when to expect delivery of certain features or how much bandwidth was left at any time during the month. These topics go back to the strategic design concept, where understanding the reasoning behind the actions or organizations is key to understanding their actions (Stickdorn & Schneider, 2012). In our context, it shows miscommunication and unclarity of goals between the continuous development team and the customer.
2. **the sales people in Futurice**, who need to understand the continuous development service's offer in order to sell it to the customers. Before the study, the continuous development service did not have a clear positioning and a clear offer. It wasn't clear what was the mission, what kind of customers they would like to serve and the attitude towards the sales people was rather reactive and reserved. This also lead to missed undiscovered opportunities due to bad flow efficiency due to long reaction times and sporadic value-adding activities (Modig & Åhlström, 2013).
3. **other Futurice employees**, with business, development or design background – for them, the continuous development team may be a next step in their career, offering them learning opportunities and a different environment than the normal service creation environment. It was noted that many times the offer to other Futurice employees was not clear, and the needs of the continuous development team itself were not clear.
4. **the teams developing new services** – the continuous development team can provide valuable feedback on how to build robust, sustainable and easy adjustable systems to the new service creation teams inside the company. In this relationship, several topics have risen, such as refactoring, symptoms of poor software design, knowledge transfer and agile application lifecycle management. These are typical

topics that the continuous development teams are facing (Martin, 2012; Hüttermann, 2012)

5. **the continuous development team members themselves**, as a learning platform and team where they could get meaningful and motivational work. Multitasking has been found as a widespread problem, also mentioned by prior research (Cohn, 2006). As the team is part of Futurice, another drawback was the perception that the Futurice culture was not extended in the continuous development team. At the beginning of the study, members of the team would go to the tech leader for guidance, but they would not collaborate between the case teams. Also, the mission of their work would be unclear. The continuous development team did not have a clear direction and it faced little support from the company, due partly to its perceived misalignment from the company's brand of creating new digital services.

The Lean Service Creation with its building blocks, and the new management paradigms covered in the theoretical framework, together with the author's observation on the team's daily work and interactions, and the team's input from the retro workshop (Appendix 2 and Appendix 3), suggest that the main challenges the team is facing can be concentrated on three areas, as below:

1. **working with the customers** - having a mutual understanding of what continuous development means, having a shared direction of work
2. **visibility of work progress** – team's flow being interrupted, and the progress slowed down by customers constantly asking when they can expect certain features
3. **team collaboration** – transparency between the projects (as the team is sharing projects, one person in more projects, one project worked upon by more people), cultivating a culture of trust and collaboration, knowledge sharing

These three areas will be evaluated in the results chapter of this report.

Based on the reframing from this chapter, the research question could be rephrased as: Looking at the topics of working with the customers, visibility of work progress and team collaboration in the continuous development teams, how might these be alleviated by LSC tools in a new generation organization?

### 3.2 Data collection

The study uses one of the principles underlying design-thinking, the abductive thinking concept, in order to investigate and understand better the research question. Abductive thinking is a type of logical reasoning that goes from observation of data to the development of hypothesis that can explain the evidence (Nixon, 2016). Abduction is a “build to think” process, and one practical manifestation of it is the “try it, prototype it and improve it” philosophy. In practice, abductive thinking may manifest itself through the experimentation and prototyping approach.

Thus, the study will use abductive thinking and suggest a set of experiments to be conducted for each problem area. Each experiment is conceived as a minimum viable way to implement the suggested improvement. The underlying assumption (discussed in previous chapters) is that the continuous development work is a natural continuation after the MVP, and thus the build-measure-learn cycle applies.

The experiments have their roots in prior theoretical concepts such as agile development, existing work such as existing LSC tools, case studies or in workshops conducted previously in the team. The experiments are built on the five principles of service design thinking (Stickdorn & Schneider, 2012):

1. User-centred: the intention is to meet the customer needs, in this case the customer being the continuous development team member. An experiment is considered relevant if it addresses one need that the target audience of the study has.
2. Co-creative: the experiment should be based on input from all stakeholders involved
3. Sequencing: the experiment should take into consideration that all the actions in the continuous development team are interrelated
4. Evidencing: the intangible aspects should be visualised in terms of physical artefacts
5. Holistic: the experiments should consider the entire environment where the continuous development team exists

Some of these experiments will be implemented with prototypes, some using existing agile tools or existing LSC tools and ethnographic research, and some by looking at existing case studies.

At the end of the chapter, the data collected will be analyzed via focus groups, ethnographic remarks and surveys.

### **3.2.1 Data collection methods**

The three methods used to collect the data have all been qualitative methods: prototyping, case studies, ethnographic research and focus group.

Prototyping “is the tangible creation of artefacts at various levels of resolution, for development and testing of ideas within design teams and with clients and users” (Martin & Hannington, 2012). For example, one LSC canvas was used as an experiment, but as it did not fit the purpose 100%, it was adapted using post its on top of existing text to be compliant with the scope of the experiment. The canvas was used in meetings with the whole continuous development team and the customer.

Case studies have been used as a research method particularly in social sciences research. The case study is a research strategy involving in-depth investigation of single

events or instances in context, using multiple sources of research evidence. (Martin & Hanington, 2012). Case studies are particularly helpful for understanding how similar problems have been solved in the past, in order to inform, compare or get inspired. They are also useful vehicles for understanding the effects of change or innovation. (Martin & Hanington, 2012). In our study, one case study involving a lack of connection between the citizens of UK and the police department was particularly interesting.

Ethnographic research has been used throughout the study, given the three roles that the author of the study had (researcher, lead of the continuous development team and service manager in customer cases). Ethnographic research refers to looking at the social interaction of users in a given environment, in order to get an in-depth view of the user's motivations, behaviours and actions. (ExperienceUX, 2017) It involves direct observation, as well as taking notes, recording, etc. For the purpose of the study, ethnographic research was conducted in different settings: while running experiments, as well as in regular daily interactions between the team members or between the team and other stakeholders.

The focus group is a discussion conducted in a selected group of people, to collect their opinions. The focus groups are most commonly known as workshops. The continuous development team has regular retrospective workshops, where the topics are: what did go well since the last time, what did not go so well, what can we improve. The workshop was conducted during summer 2017.

### **3.2.2 The basis for choosing the experiments**

As mentioned in the study setup and premises of the study chapters, the experiments need to tackle the confluence of the following areas:

- Problem areas: working with customer, visibility of work, team collaboration and cohesion
- LSC mindset
- Team as customer-value-driven, highly engaged.





and deciding which items are a must to be done, which ones should be done, which ones could be done, and which ones will not be done.

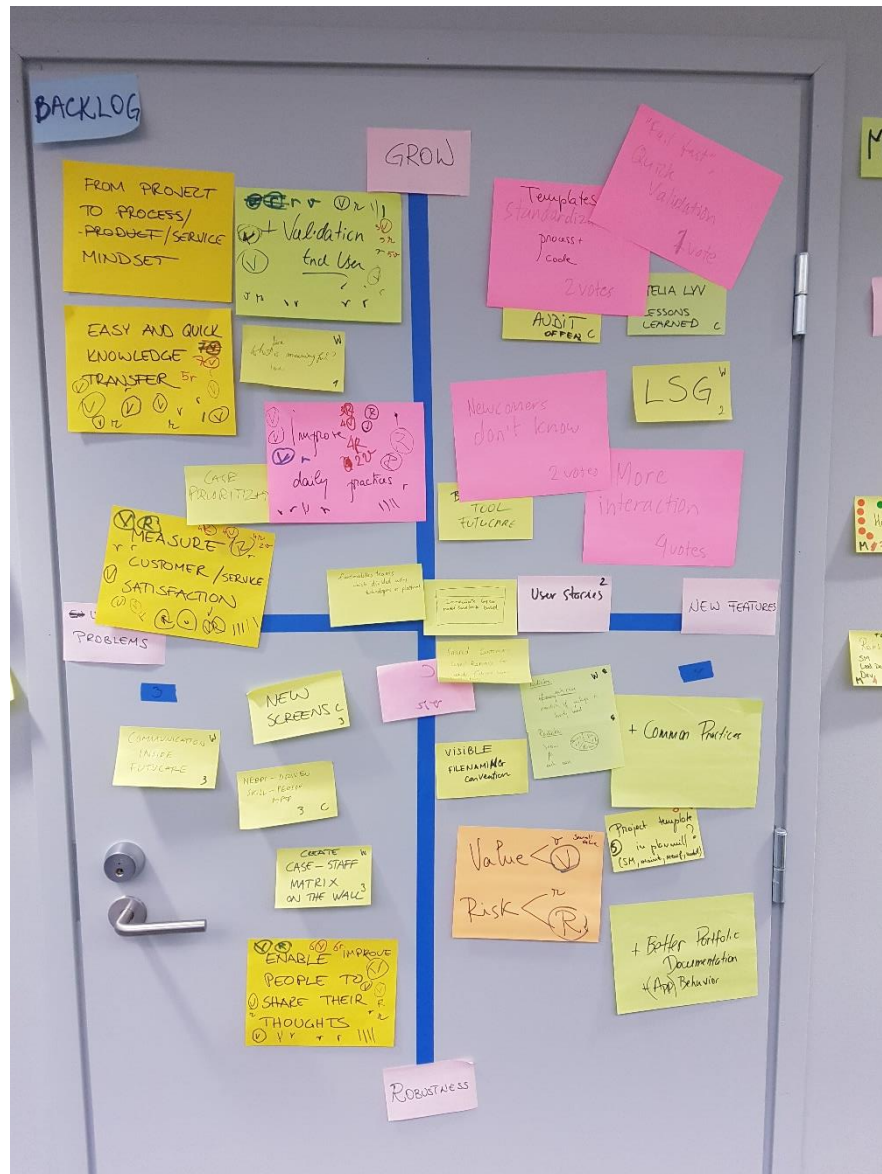


Figure 35. Team retro workshop outcomes. (Tauciuc, 2017).



Figure 36. Team retro workshop outcome, sorted. (Tauciuc, 2017)

### 3.2.3 Overview of experiments chosen





































The experiments have been chosen based on:

- The LSC mindset
- The outcome from the team retros
- Known agile development tools
- Employee engagement principles, coming from traits of high performing, customer-value driven, agile teams.


The backdrop of the experiments has been the three key breakthroughs of the teal organizations, that Futurice aspires to: self-management, wholeness, evolutionary purpose (Laloux, 2014).

Below is an overview of the experiments chosen, together with the motivation for each and the data collection method used. The following chapters explain each experiment in detail.


## Experiments overview

Problem area	Experiment	Experiments are based on...	Data collection method used is...
Work with customer	Business objective canvas	  	
	Focus canvas	  	 
	Service manager in case team	  	 
Visibility of work progress	Story points, velocity	 	
	Work in sprints	 	
Team collaboration and cohesion	Time radiator	  	
	Cross-teams daily scrums	  	
	Whole team weeklies	  	
	"A star and a wish"	  	


Experiments are based on...




LSC mindset



Team retro outcome




Agile development




Employee engagement


Data collection method used is...



Prototyping



Ethnographic research



Case study

Figure 37. Overview of the experiments chosen for the study. (Tauciuc, 2018)

The experiments have been run between April 2017 and November 2017. The experiments do not have interdependencies, and they have been run concurrently.

### 3.2.4 Business objective canvas experiment

The business objective canvas experiment refers to validating the suitability of the original LSC business objective canvas, to be used in the post – MVP launch phase and ethnographic research has been used to collect relevant data.

Problem area	Experiment	Experiment is based on...	Data collection method used is...
Work with customer	Business objective canvas	  	

Figure 38. Business objective canvas experiment, motivation and data collection method. (Tauciuc, 2018)

This experiment has been chosen because:

- **LSC mindset** – The business objective and context canvas has been used in LSC in the initial discussion with all the stakeholders. It sets a mutual understanding on the purpose of the service aimed to be built and the motivation behind it (strategy fit with the customer). It also opens the discussion about aiding and restricting factors for the service and it sets success metrics. In other words, the canvas helps

the stakeholders see the connection between the service to be built and the business outcome. This is important post-MVP launch as well, since the MVP is only the minimum viable product, and more features are added on top of it. These features need to be in line with the business objectives of the customer. Also, because the service evolves continuously, there needs to be a scalable approach to the requirements (Leffingwell, 2007) and this is enabled by understanding the business objective. Some examples of business objectives could be: maintaining/increasing profitability, increasing return on investment, improving market share, superior brand recognition etc.

- **Team retro outcome** – the continuous development team has been mentioning that often there was no clear purpose of the work and the roadmap was unknown.
- **Employee engagement** – one pain point of the continuous development team has been the constant need to explain to the customers why the continuous development is important and what is the content of the work. This can be alleviated by making more visible the connection between the development and the increase in business value of the service and adopting a customer focus, rather than feature focus, approach. In order to be customer-value driven, the team needs to understand what “value” means for the customer. Once this happens, the team is more empowered and can take more accountability on the service. The canvas helps with all these aspects.

Prior research sustaining this experiment includes:

- increase in customer willingness to invest in more functionality when the link to the business value is apparent (Schwaber & Beedle, 2002)
- stakeholder focus could be focus on the business perspective, technology perspective, customer perspective (Berkun, 2005)

## BUSINESS OBJECTIVE AND CONTEXT

Fill together with the person funding this project

<b>How will we know that we have succeeded?</b> (After a month? After one year?)	<b>Who needs to be involved?</b> (stakeholders, old and related projects...)
<b>What is our business objective?</b> (business problem/need/opportunity...)	
<b>What do we need to take into account?</b> (Brand strategy...)	<b>Why is it important?</b> (How does it fit our strategy?)
<b>What restricts us?</b> (Budget, Schedule, Organization, Legal, Current business...)	

**ASK WHY.**

**futurice** **LEAN SERVICE CREATION**

Have we filled this together with the person funding this? ☐  
 With our supervisors? ☐  
 Is this clear and understandable? ☐  
 Are we comfortable to proceed? ☐

Lean Service Creation Toolbox by Futurice Ltd is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Figure 39. The business objective and context canvas. (Futurice, 2013)

The canvas has been used in the kick-off meeting with one of the team's customers from the telecommunications field. The MVP of the service was done before in the company, and the continuous development team started working on it once the MVP was launched. In the initial meeting between the continuous development team and the customer, the first item on the agenda was to go through the canvas and open the discussion.

After it was filled in, the canvas was included in the meeting notes and added as a reminder to all consequent monthly meetings between the stakeholders.

### 3.2.5 Focus canvas experiment

While the business objective canvas tackles the continuous development team's work at a business value level, the scope of the focus canvas is to get a mutual understanding on the type of daily work the customer finds valuable.

Problem area	Experiment	Experiment is based on...	Data collection method used is...
Work with customer	Focus canvas	  	 

Figure 40. The focus canvas experiment motivation and data collection method used.

(Tauciuc, 2018)

There was mismatch observed between what the customer wanted (for e.g. bug fixing) and what the continuous development team assumed the customer wanted (for e.g. a new feature). Also, in many cases, the team had good suggestions about improving the scalability of the service or implementing ways to keep the service relevant in reaction to changes in environment / user base / platforms. Those discussions were difficult and stressful for the team, because in some cases the customer had different priorities, or the changes were not aligned with the strategy the customer had for the service. Thus, the team needed a tool to facilitate these conversations and bring transparency.

The focus canvas is based on LSC's existing *Weekly LSC board* canvas, more specifically on the section called *we feel that currently our work is driven by*:

## WEEKLY LSC BOARD

- Ensure Lean Service Creation during the building phase

**RIGHT NOW WHAT IS THE MOST CRITICAL...**

**TECHNICAL ISSUE**

**BUSINESS QUESTION:**

**END USER ISSUE:**

**VALIDATION**

What is validated by the work we are now doing?

**WEEKS BIGGEST FEAR / LIMITATION:**

**WE FEEL THAT CURRENTLY OUR WORK IS DRIVEN BY:**

**BUSINESS PROBLEM**


**TECHNOLOGY**

**END USER PROBLEM / INSIGHT**

**PRODUCT**

**GUESTBOOK**

Team space visitors + date:


 Lean Service Creation Toolbox by Futurice Ltd is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



**LEAN SERVICE CREATION**

Figure 41. Weekly LSC board canvas. (Futurice, 2017)

Applying the same principle of what would drive the continuous development work, the team came up with the following factors:

- fixing issues that current users have (sustaining the current user base)



- creating new features (growing the service)
- keeping up with the technological changes, environments and tools upgrades, re-factoring, increasing the level of maintainability of code (building a robust system)
- focusing on keeping up and growing the relevancy and business value of the service (following up the business objectives)

The first iteration of the focus canvas can be seen below:

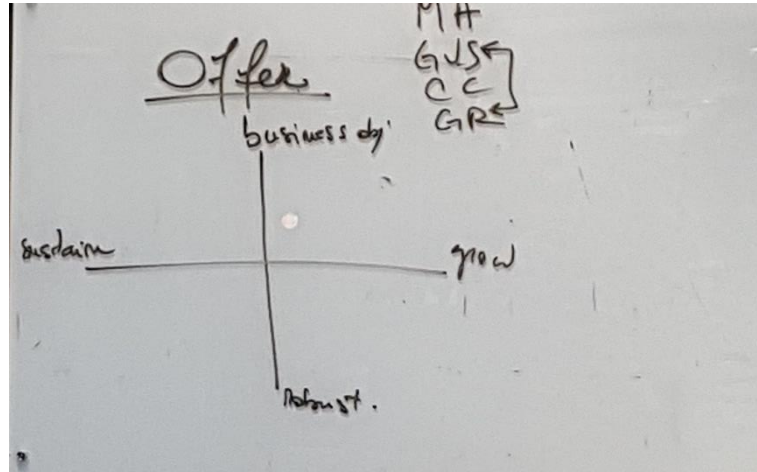


Figure 42. First iteration of the focus canvas. (Tauciuc, 2017)

This canvas is meant to be used in the planning meetings with the customer (the kick off and periodically once in 3-4 months, depending on how actively the customer wants the service to be developed).

In practice, the continuous development team asks the customer: For the next three months, what is the most important for you?

- To build a robust system – this is usually the option the customers are interested in when the service has been just launched, or when they want to keep gathering data about the service performance. The customers can choose to fix bugs (sustain) or to implement new features that the current customer base is asking for (grow).
- To follow up the business objective – in this case, the sustain / grow choice is infused with a more aggressive approach on tackling the business value of the service

Having the clarity on what the customer value means, the team can now focus and adjust the work based on it. This brings the team closer to a customer-value driven team (Moreira, 2017) and increases its efficiency as it focused on value-adding activities (Modig & Åhlström, 2013). It also gives the team more autonomy and motivation, by understanding and helping shape the mission of their work.



This experiment has been applied in two customer cases, both from the telecommunications field. The discussion was held two times with one of the customers, and once with the second customer.

To collect relevant data about the success of the experiment, two methods have been used:

- Ethnographical research, as the author of the study was heavily involved in the experiments in her role as service manager and lead of the continuous development service
- Case study. Macaulay mentions a case study named “A systems approach to housing repairs” by J. Selddon and B. O'Donovan (Macaulay, 2012). In this case study, one problem that the tradesmen had was about the pressure to apply quick fixes, short-term patches rather than investigate the root cause of the problems and apply a proper repair. In a similar fashion, the continuous development team did not get enough understanding from the customer when they talked about the important of building robustness into the system and were asked for quicker solutions. In the case study, the tradesman undertaking the work was allowed to decide the best way to do it, unburdened by management, and the quality of the repairs increased. Just like in the case study, the focus canvas aims to give more freedom to the team, because all stakeholders have the same understanding on where the efforts should be concentrated.

### 3.2.6 Service manager in case team

The overview of the experiment can be seen in the figure below:



Figure 43. The service manager experiment motivation and data collection methods.  
(Tauciuc, 2018)

The key role of a service manager is to ensure that the needs of the customer are satisfied. This is done by constant communication with the customer and the team, and by having a deeper understanding of the customer's vision and context and helping build the roadmap towards the vision.

At the beginning of the study there were 7 micro teams working on customer cases. In most of the cases, the service manager was a person outside the team: usually the account owner or another sales person.

The team was impacted by the distance between them and the service manager. Consequently, the team was the one holding the conversations with the customer on a daily basis. Having a person outside the team meant that the team needed to spend more time

synchronizing the information with the service manager, while not always having all the business-related information needed for their daily work. Also, the team mentioned disconnection from the service manager as a factor that slows down the progress in the development.

Having the service manager inside the case team means that the role of service manager is shared by the developers forming the continuous development team dedicated for the customer case. Bringing the service manager role inside the team increases the team's accountability, which is an important aspect in building an adaptive organization and building innovation (Ries, 2011). Also, it shortens the learning cycle, and provides greater decision-making authority to the teams. The team has more space to learn and experiment, and this way uncovering ways in which to bring more value to the customer (Gothelf & Seiden, 2017). This approach supports also building up a teal organization, by promoting the self-management aspect via self-organizing teams and radically simplified project management (Laloux, 2017).

The experiment took place in two phases: in the first phase, the service manager role was moved inside the continuous development team, and in the second phase, the role was moved further in the customer case micro teams.

## SM in customer case experiment



Figure 44. Moving the service manager role inside the customer case teams. (Tauciuc, 2018)

The reason for the two phases lies in the perception that the company (and the service managers and sales people outside the team) had on the continuous development team. There was a disconnection between these stakeholders: the continuous development team wanted to be able to take a more active role in the customer relationships and have

a greater influence on their work, and they were missing this communication with the service manager. In an analogous manner, the service managers were not aware of the entire potential that the team has and were not ready to provide more autonomy to the teams. This is a comparable situation as the one mentioned by Stickdorn & Schneider in the Mypolice and Snook case study (Stickdorn & Schneider, 2010). There, the problem was that the citizens of UK did not feel listened to by the police – just in the same way that the continuous development team members felt they did not have the space to have a bigger impact on the project. In a similar fashion, just as the police has in many cases lost the confidence of the British public, the service managers outside the team felt that they need to be present in the relationship with the customer, because the team needed their support. In the case study, the situation was solved by creating an independent and neutral space that could “close the feedback loop, fostering constructive and collaborative conversations between the public and the police” (Stickdorn & Schneider, 2010). In a similar fashion, the independent space was, in our study, the business-skilled team members from the continuous development team, who were freshly recruited in the team and thus there was no pre-established perception on them. In the first phase, these persons took the role of service manager in the customer cases.

The next step was to diffuse the role in the micro team dedicated per customer. This was piloted in two customer cases, with 2 – 4 people micro teams.

### **3.2.7 Work in sprints**

Unlike the teams working on creating a new service, the continuous development team did not have the same feeling of accomplished whenever a new feature was implemented. The work was not structured, but rather an extensive list of tasks to be done, with no clear milestones other than the end of the time allocated per month. This brought down the morale and damaged the cohesion between the continuous development team and the other development teams in the company.

In practice, as continuous development is software development, the team needed to increase the visibility of the work and bring the same agile approach that the project teams had. The scope was to create micro teams that “work as one” (Cohn, 2006). Thus, working in sprints has been experimented.




Problem area	Experiment	Experiment is based on...	Data collection method used is...
Visibility of work progress	Work in sprints	 	

Figure 45. Working in sprints during continuous development. (Tauciuc, 2018)

This experiment was applied to three customer cases, and the iteration varied between 2 weeks and 1 month. The focus of each iteration was a business priority agreed upon with the customer based on the focus canvas, and the purpose was to have functionality ready to be released.

### 3.2.8 Story points, velocity

One challenge that the continuous development team faced was the difficulty of answering to the customer when a certain feature would be ready, or what a certain delivery would contain. In the context of a high-speed project, the answer is foreseeable. However, in the context of continuous development, the team needs to spend the limited time per month not only on the items in the backlog, but also on any potential issues related to the robustness of the system. Thus, the necessity of experimenting the usage of story points and velocity in the context of known work-to-be-done and unforeseeable work has risen.

This experiment is related to bringing more visibility in the work of the team.




Problem area	Experiment	Experiment is based on...	Data collection method used is...
Visibility of work progress	Story points, velocity	 	

Figure 46. Story points experiment motivation and data collection method. (Tauciuc, 2018)

The usage of story points is a common practice in agile projects. “Story points rate the relative work, risk and complexity of a requirements or story.” (PMI, 2017). Story points are a unit of measure to express the overall size of a piece of work. While the value itself is not important, the relative size between the tasks is important, and that is the one that is reflected in the points. (Cohn, 2006). The values that were used in this experiment are S (small, task may take 1-2 hours), M (medium, task may take up to 4 hours), L (large, task may take a day).

The velocity gives a measure of the team's speed of progress. It is calculated by summing up the points for all the tasks completed by the team during the iteration or sprint (Cohn, 2006). The key role of the velocity is to allow the team to use historical data in order to plan accurately the work (PMI, 2017). At the end of each iteration (sprint), the team would calculate the velocity. The information was stored and used to estimate what tasks, and how much, could be done in the next iteration.

The approach was tested in two customer cases.

### 3.2.9 Time radiator

The last category of experiments is focused on building team collaboration and cohesion. The first experiment is the time radiator:

Problem area	Experiment	Experiment is based on...	Data collection method used is...
Team collaboration and cohesion	Time radiator	  	

Figure 47. The time radiator experiment. (Tauciuc, 2018)

The idea of the experiment came from the team retro workshop. One problem that the micro teams' members had was about splitting the allocated monthly time so that they do not exceed the monthly limit. As the team members do not have visibility in the way each other reports the hours they worked, there was constant asking about the remaining time available. In several situations, more time was used than available. The daily work process was made heavier.

Thus, the team has decided to implement a read-only view of the time usage per customer case, and the view would be updated on an hourly basis. For each customer case, the following information would be included:

- name of customer case
- amount of planned time per month, where applicable
- color-coded time spending
- names of team members in the micro team

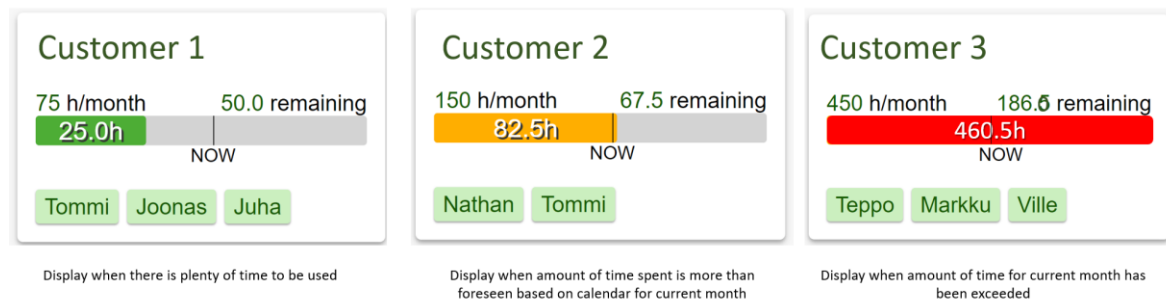


Figure 48. Examples of display for different situations regarding the time already spent per case. (Tauciuc, 2018)

The experiment was run for all customer cases, and the information for all cases was displayed in the same view. The information was visible on a screen mounted in the same area where the team was sitting.

This experiment sustains building of a teal team, by managing crisis using transparent information sharing and collective intelligence. It also sustains the information flow, by making the time related information available in real-time to everyone interested. (Laloux, 2014).

### 3.2.10 Cross-teams daily scrums

The next practice introduced as experiments has its roots in the agile practices:

Problem area	Experiment	Experiment is based on...	Data collection method used is...
Team collaboration and cohesion	Cross-teams daily scrums	   	

Figure 49. Cross-teams daily scrum experiment characteristics. (Tauciuc, 2018)

One problem area that the team mentioned in the team retro has been related to knowledge sharing between the customer cases, and the occasional lack of support between developers with the same background. As the micro teams dedicated per customer cases are rather small (2-4 people), there were times when for e.g. there was only one android developer in the team. This developer felt the need for sparring support and greater collaboration with the other android developers from other micro teams. The developers wanted a smoother knowledge transfer as well, and socialization and externalization have been found by Nonaka & Takeuchi to be the best ways to transfer tacit knowledge.

From \ To	<i>tacit</i>	<i>explicit</i>
<i>tacit</i>	<b>Socialization</b>	<b>Externalization</b>
<i>explicit</i>	<b>Internalization</b>	<b>Combination</b>

Figure 50. Four modes of Knowledge Conversion, Nonaka & Takeuchi  
(<https://newtrendsmanagement.wikispaces.com>, 2011)

As the aim was to have agile teams, one natural practice to be introduced was the daily scrum, but with a twist. Instead of having customer case specific daily scrum, the meetings would be held between continuous development team members that have the same background (e.g. android, mobile, full stack).

Prior theoretical research suggests that this approach would reduce the silo mentality, increase collaboration inside the team and create a culture of continuous learning – all relevant in the context of a sense&respond organization (Gothelf & Seiden, 2017). The experiment would also increase the happiness of the team members, as “when people are working on a creative project, they’re happy. When a team can come together around a creative cause or a knotty problem, they want to come to work every day” (John Maeda cited by Martin, 2009, p132). And not lastly, a tighter collaboration would bring cross-pollinating of ideas from different customer cases, that is part of designing a service for growth (Ingle, 2013).

The experiment was done with the team members that had as core skill either android, either iOS programming. The experiment affected all customer cases where these team members were involved.

The data about the experiment has been gathered by ethnographic research, from the three perspectives that the separate roles of the author of the study offered.

### 3.2.11 Whole team weeklies

The “whole team weeklies” experiment has the aim of improving the team collaboration and cohesion by providing a space where everyone in the team can share freely their thoughts, opinions, questions, etc. The idea of the experiment is to have a meeting, once

per week, for one hour, with the whole continuous development team, regardless of the customer case they are working on.

Problem area	Experiment	Experiment is based on...	Data collection method used is...
Team collaboration and cohesion	Whole team weeklies	   	

Figure 51. The "whole team weeklies" experiment. (Tauciuc, 2018)

The meeting would be held on Fridays in a less formal setting (for e.g. the informal space next to the sauna in the office), where everyone could be seated in a circle and see each other. The flow of the meeting was set up as following:

1. Smileys: each person in the team would mention how they have been feeling the past week by using a color (green = everything is well, yellow = some reason for concern, red = serious reason for concern that calls immediate action). Each person would then proceed to talk about main highlights during the week, be them professional or personal.
2. Information on other ongoing activities related to the continuous development team, possible customer cases etc.
3. Financial information on the performance of the service – this section was added later after some iterations of the weekly meeting

The experiment was started during summer 2017 and it has been ongoing every Friday since then.

### 3.2.12 "A star and a wish"

The prerogative of the experiment "a star and a wish" was to build collaboration, empathy and encourage individual growth.





Problem area	Experiment	Experiment is based on...	Data collection method used is...
Team collaboration and cohesion	"A star and a wish"	   	

Figure 52. The "a star and a wish" experiment. (Tauciuc, 2018)

The experiment refers to a session of feedback in pairs. The team members chose their own pair, based on how much they worked together or how well they knew each other. Once the pairs were established, each person received a paper as below and had five minutes to answer each question.





What I appreciate most about you is \_\_\_\_\_

Like when \_\_\_\_\_

AN APPRECIATION OF A BEHAVIOUR THAT SOMEONE ALREADY EXHIBITS



What I'd like to see more of from you is \_\_\_\_\_

[Expected impact] \_\_\_\_\_

A SUGGESTION AS TO WHAT SKILLS AND BEHAVIOURS TO BUILD ON

A star and a wish – side A, positive feedback

A star and a wish – side B, constructive feedback

Figure 53. "A star and a wish" experiment. (Tauciuc, 2017)

The “star” refers to positive feedback for the receiver, including a concrete example when the behavior was apparent. The “wish” refers to a behavior that the giver of feedback would like to encourage or instill in the receiver, including some examples of impact this behavior would generate. The purpose of this exercise was to provide concrete feedback, sustained by concrete situations, that the receiver can build on.

Offering and receiving this kind of feedback has been something new in the team, as usually the team members would receive feedback from own supervisor, who was not involved in the continuous development work.

The experiment was run once.

### 3.3 Data analysis

The previous chapter described the practical experiments that have been conducted in order to collect relevant data. The experiments touched all the areas of interest from the team’s challenges point of view: work with customer, visibility of work progress, team collaboration and cohesion. The data collected has been analyzed using quantitative methods such as surveys, and qualitative methods such as focus groups and observations (as part of ethnographic research). In some cases, cognitive mapping was also used.

#### 3.3.1 Data analysis methods

The analysis of the data has been done with three main methods: surveys, focus groups, observations and cognitive mapping.

Surveys are a quantitative method for “collecting self-reported information from people about their characteristics, thoughts, feelings, perceptions, behaviours, or attitudes.” (Martin & Hanington, 2012). Surveys have been used three times for the duration of the study: in June 2017, in August 2017 and in January 2018. The main scope of the surveys was to follow up the continuous development team’s pulse, such as empowerment, the feeling of trust in the team, openness, enjoyment of work, level of challenge etc.

The June 2017 survey was conducted when the experiments were being discussed, they have not been implemented yet. For the August 2017, some surveys have already been running, and in January 2018 the experiments’ official trial period has ended (even though some were still running, as being incorporated in the way of working and not perceived anymore as experiments). The January 2018 survey contains extra questions related to a retrospective of 2017. The invitation to the January 2018 survey can be seen in Appendix 6 and the list of questions for all surveys can be seen in Appendix 6-8.

Another survey being used in Futurice is the OfficeVibe. OfficeVibe measures employee engagement, and it includes topics such as personal growth, ambassadorship, relation with peers, relation with managers, recognition, feedback, wellness etc. With the results from OfficeVibe, we can compare the results in the continuous development team against the overall Futurice results.

Another data analysis method used has been the focus group. A focus group involves a carefully selected group of participants (between 5 and 10), sitting in a comfortable environment, that have a facilitated discussion around certain topics. The purpose of a focus group is to collect the participants’ opinions and enable further decision making. (Interaction Design Foundation, 2002). For this study, a focus group was held in January 2018, with the purpose to have a retrospective discussion on 2017: what are the things that went well, what are the learnings, what could be done better.

Throughout the duration of the study, observation has been used by the author of the study in different settings and from her three different perspectives: as lead of the continuous development service, service manager in customer cases and researcher. The study offered several opportunities for observation, also during discussions with team members and other stakeholders. As Futurice is a company based on trust and on the 3x2 principle described in the first chapter, observation has been a very valuable tool. Observation served analysing the impact of the experiments on the team and on the wider environment and finding ways to adjust and learn in order to calibrate the efforts towards the desired goals.

The last method used for analysing data has been cognitive mapping. Cognitive mapping uses visualization to make sense of a problem space. In a similar fashion to mind mapping, cognitive mapping is most effective when used to structure complex problems and to inform decision making. (Martin & Hanington, 2012). In practice, cognitive mapping has been done regularly during the study, in different situations such as: discussions within different working groups about the evolution of the continuous development offer and service, 1-to-1 discussions with different team members, weekly meetings for synchronization between the continuous development team lead and different other stakeholders.

Below is a summary of the data collection methods, and how they were applied in practice in the study.



Figure 54. Summary of data analysis methods used for the study. (Tauciuc, 2018)

### 3.3.2 Team survey analysis 2017 – 2018

The first set of interesting data can be observed by analyzing the data via the survey results. In June there were 14 respondents, and 9 respondents in August and January respectively. Below is the comparison between the answers given to the shared questions of the team survey in June 2017, August 2017 and January 2018:

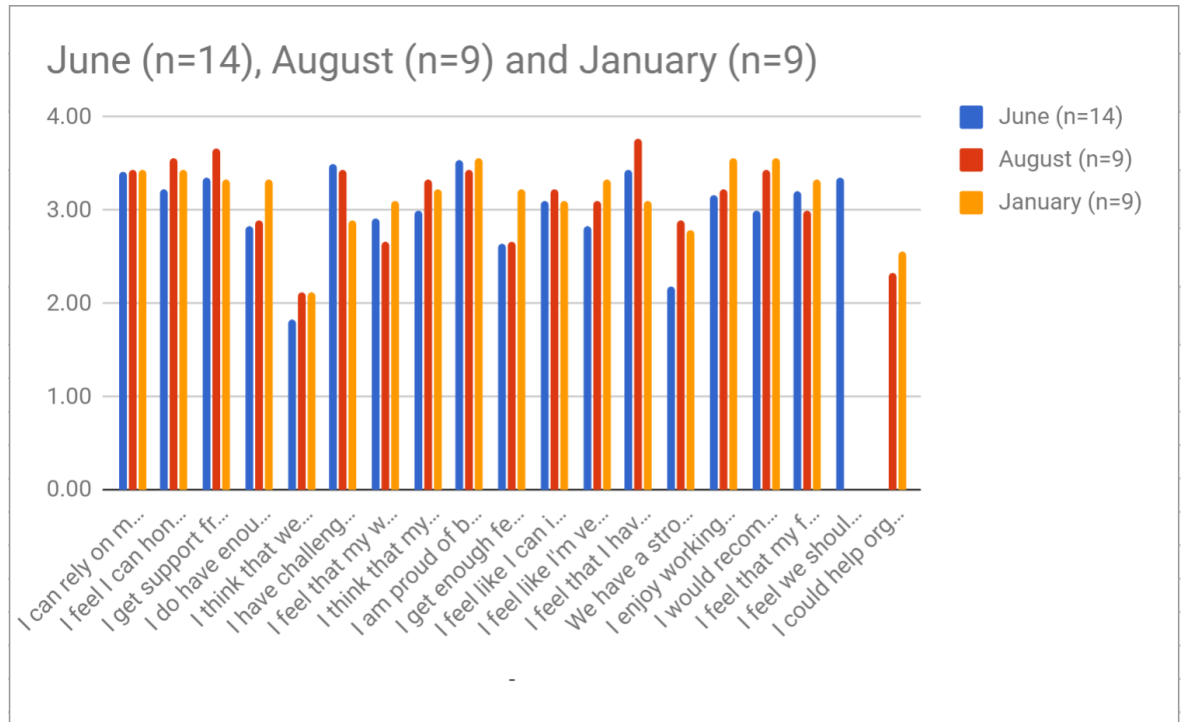


Diagram 1. Comparison on team survey answers. (Tauciuc, 2018)

Each question from the diagram above could be answered on a scale from 1 to 5, where 1 = not at all, 5 = absolutely. In the survey questions, “Futucare” refers to the Futurice internal name of the continuous development team.

The following questions have been having their minimum score in June, and registered a steady increase until January:

1. I can rely on my colleagues’ skills and abilities
2. I do have enough visibility and information about what is going on in the Futucare team
3. I think that we share enough practices between cases
4. I get enough feedback about my work
5. I feel like I'm very good at my job
6. I enjoy working at Futucare
7. I would recommend Futucare team for other Futuricians

	I can rely on my colleagues' skills and abilities	I do have enough visibility and information about what is going on in the Futucare team	I think that we share enough practices between cases	I get enough feedback about my work	I feel like I'm very good at my job	I enjoy working at Futucare	I would recommend Futucare team for other Futuricians	Average
June (n=14)	3.42	2.83	1.82	2.64	2.83	3.15	3.00	2.81
August (n=9)	3.44	2.89	2.11	2.67	3.11	3.22	3.44	2.98
January (n=9)	3.44	3.33	2.11	3.22	3.33	3.56	3.56	3.22
January - June difference	0.81%	15.00%	13.88%	18.18%	15.00%	11.30%	15.63%	12.70%

Figure 55. Questions that had a steady increase in score during the study. (Tauciuc, 2018)

The questions above had their minimum score before the experiments were running, and have their maximum score in January 2018, when the experiments were not running anymore. This shows that some experiments had a lasting impact, or they became part of the daily way of working approach and not seen as experiments anymore. The scores have increased with a minimum of 0.81% ("I can rely on my colleagues' skills and abilities) and a maximum of 18.18% ("I get enough feedback about my work"). On average, the score for these questions increased with 12.70%. The difference shows notable increase in the respective questions and underlying themes.

The questions above have the recurring themes: collaboration, trust, empowerment, ownership, safety, motivation. This shows that the team members have felt much more comfortable to talk to each other, they felt they collaborated more, and got a sense of accomplishment and being proud of their work. While these scores are not influenced only by the experiments introduced, the experiments had a strong influence on the scores.

It's interesting to note that all themes are traits to be found in customer-value driven and/or high performing teams. This shows that while the team dynamics have been improved, the team also became more customer focused and the performance increased.

There are also questions that showed some variation during the study period, and had the maximum score in August 2017, when all experiments were ongoing. These questions are:

8. I get support from my colleagues when I need it
9. I think that my current project/work tasks are interesting
10. I feel like I can influence things at work to improve things around me
11. I feel that I have possibilities to improve my professional skills in my work
12. We have a strong sense of togetherness in Futucare team

		I get support from my colleagues when I need it	I think that my current project/work tasks are interesting	I feel like I can influence things at work to improve things around me	I feel that I have possibilities to improve my professional skills in my work	We have a strong sense of togetherness in Futucare team	Average
June (n=14)	3.36	3.00	3.09	3.44	2.18	3.01	
August (n=9)	3.67	3.33	3.22	3.78	2.89	3.38	
January (n=9)	3.33	3.22	3.11	3.11	2.78	3.11	
January - June difference	-0.71%	6.90%	0.65%	-10.71%	21.45%	3.09%	

Figure 56. Questions that registered a maximum score during the experiments. (Tauciuc, 2018)

The following aspects were influenced by an increase in score during the experiments: collaboration and support, enjoyment of work, feeling of empowerment and ownership, togetherness. When the experiments were over, only 2 questions registered lower scores than before the experiments: getting support from the colleagues and having the possibility to improve own professional skills. One explanation may come from having a holistic view on the team situation. The end of 2017 has been a very busy period for the team, and everyone has been working in different customer cases, having less flexibility and thus less available time to invest in any activities besides the customer case work. Also, the cross-teams daily scrums were not running anymore, and this influenced the feeling of support inside the team. In the same time, the feeling of togetherness in the team increased with 21.45%, so the reason why the team members felt they did not get enough support is not because of lack of intention, but rather from other external causes (such as lack of time). Even if the maximum scores were in August, overall in January there was an increase of 3.09% compared to June when the experiments were not running.

The last category of questions are the ones that recorded the minimum score during January 2018. These show areas that have a decreased level once the experiments have been running:

	I get support from my colleagues when I need it	I have challenged myself (tried new things, learned new things, shared my knowledge etc.)	I feel that I have possibilities to improve my professional skills in my work	Average
June (n=14)	3.36	3.50	3.44	3.43
August (n=9)	3.67	3.44	3.78	3.63
January (n=9)	3.33	2.89	3.11	3.11
January - June difference	-0.71%	-21.15%	-10.71%	-10.37%

Figure 57. Questions with a minimum score in January 2018. (Tauciuc, 2018)

It is interesting to note that two of the questions (“I get support from my colleagues when I need it” and “I feel that I have possibilities to improve my professional skills in my work”) showed their maximum score during the period when all experiments were showing. Having a lower score in January may show that those experiments need to be identified, studied and incorporated in the way of working as such or with a modified version.

The only question that showed steady decrease in score is related to how challenging the work has been. The right level of challenge increases the motivation and the sense of accomplishment, and this topic needs to be addressed as a recommended next step. The decrease in score may be related to an uniformization of practices among customer cases, and thus the team members may need a different type of challenge in addition to the work itself.

Overall, the average score has increased in January 2018 compared to June 2017:

	Average
June (n=14)	3.01
August (n=9)	3.17
January (n=9)	3.20
January - June difference	6.02%

Figure 58. Overall score changes between June 2017 and January 2018. (Tauciuc, 2018)

The figure above shows that while there is some variation between the question, the overall situation improved steadily, and the effects of the experiments have been long lasting, their influence being seen also when the study's period was over. Based on that, we can conclude that the experiments have brought a change in the mindset of the team and stakeholders, and they have not merely been a set of tools.

### **3.3.3 Team retro 2017 analysis**

The team survey held in January 2017 contained two parts: one part has been analysed in the previous chapter, as it contained the questions asked before in June 2017 and August 2017. The second part of the survey referred to an overview of 2017, and it is best to be analysed in relationship with the outcome of the team focus group retro held in January 2017. For a comprehensive list of the answers to the survey please refer to Appendix 10, and for the notes of the focus group please refer to Appendix 11.

The purpose of the focus group was to provide an open forum for discussion, and to verify what experiments did the team remember from 2017. Based on the notes, the focus group was held a bit too late after the experiments ended, and many of the team members remembered only some of the experiments (e.g. weeklies). The main areas that the team members mentioned were:

1. Overall on 2017, the work amount has been quite high. This left little flexibility for the employees to focus on learning new technologies and on sharing practices. However, the cross-teams daily scrums (called the android dailies) helped.
2. The team expressed a strong desire to work with newer technology and came up with ideas on how to incorporate this wish in the work, by updating the offer of the service to include digital transformation from older technologies to newer ones).
3. The team also wanted a stronger feeling of "team", as they felt that the continuous development team is rather a collection of micro teams that work together on customer cases. One of the underlying reasons of this wish was to have a tighter network to provide and receive help across customer cases. This could be tackled with extending the cross-teams daily scrums to other expertise as well.
4. The team also decided to have periodical retrospectives within the customer cases – this is a core practice of LSC. Based on the fact that the periodical retrospectives were not mentioned during the study, the team's decision can be considered as proof that the team mindset has been changed even more towards LSC during the study.

The comments above can be explored further using the answers the team members provided to the part 2 of the survey from January 2018. The main purpose of the part 2 was the same as with the focus group – to get an understanding on the impact of the study on 2017. While the focus group was based on the interactions and discussions in the group, the survey offered the chance for self-reflection and deeper questions.



Just like in the focus group, the team mentioned as impediments in their daily work: being too busy and too little knowledge sharing. These needs are to be addressed by changes in the teams (already planned during autumn 2017). Even with these difficulties, the team members describe 2017 as a year when the team's journey could be characterized as: "interesting and eventful", "very noticeable growth", "challenged, busy, interesting", "teaming, communicating, doing". These are also pointers to what the team values: the team members want to be challenged, what the change, and are motivated by teaming up. Receiving more recognition inside the company has been another item that was mentioned in the answers.

When asked about what new practices were tried out in 2017, referring to the experiments, the team mentioned the weeklies, the daily scrum, supporting each other, changing the location inside the office and helping each other more. No other experiment was mentioned. There can be different reasons for this, amongst which the fact that some experiments have been incorporated in daily work and have not been seen as novelty anymore, other experiments have been ran too few times, the team members who answered the questions were not involved in those experiments or there was too much time between the experiments and the time of filling in the survey. However, when asked specifically about certain experiments, they could assess the impact on their daily work:

5. **LSC canvas usage** was remembered by 2 /14 respondents. Because the experiment was tried in very few customer cases where few team members were involved, the result is not surprising. The impact of the canvases is mentioned as: "clarified business situation", "they supported the clients to think over". LSC canvas is also mentioned as something to try in 2018, and also as a way to make the meetings with the customers more effective by setting the focus on important matters in both short and long term.
6. **The role of the service manager** was seen as an interface between the clients, teams and business stakeholders. Only one respondent mentioned a preference towards someone in the development team being also the service manager. Everyone else preferred someone with a business background to act as the service manager, to "protect developers". However, all comments mentioned a strong collaboration between the service manager and the team, and the need to customize the approach based on the customer case situation and demands.
7. 7/10 respondents mentioned having used the **story points**, but only 2 mentioned measuring the velocity. When it was measured, it had a positive impact as the task progress improved. When the velocity was not measured, the story points were used to divide the tasks in smaller ones to favour the progress of the development and more accurate estimations.
8. 3 respondents mentioned having **worked in sprints**. The main reasons were to keep a reasonable work balance for the team and defining the pace and the goals of the work. In the cases where working in sprints was not applied, the respondents mentioned "sprints aren't that useful when mainly responding to problems in services (which are unpredictable)", "we implement one task and move to the next one when the first one is ready", "we could have used Kanban with periodical checkpoints with the same results". Having more sharing sessions between the

teams would help equalizing the way of working for getting the best results. Also, in some customer cases having development goals set for two weeks may not be the most suitable approach.

9. **The time radiator** was an experiment that everyone agreed upon as having been very useful: “easy to get knowledge on the status of different cases at a glance”, “it creates visibility on the pace”, “it helps to choose what to work on”. While the initial scope of the radiator was to create visibility on the amount of time left to be spent, the experiment proved to be much more useful. Based on the answers, the time radiator brought more transparency and a sense of progress, while also acting as a planning tool.
10. Few respondents, 2 out of 9, mentioned having taken part in **the cross-teams daily scrums** (the android dailies). Due to changes in the team and the timing of the survey, only two people from the continuous development team had mobile background at the time when the survey was filled in. Based on this information, all team members with mobile expertise were part of the experiment and enjoyed the experiment. One respondent mentioned that he “enjoyed when people had those. It helped them start sharing more”, even though he was not involved himself. This attitude proves that the experiment could be expanded to other expertise. Also, in the survey, the experiment was mentioned as a practiced to be continued in 2018, for other sub teams as well.
11. The **“a star and a wish”** experiments received favourable comments, such as “it was nice to hear all these things”, “feel grateful to have the person by your side”, “very good discussion”. Some respondents expressed a wish to continue the practice and hinted that it could be used as a platform to understand the needs that are in the team and act on this understanding.
12. The **weeklies** experiment was another practice that received full support from the team. The team members appreciate the smileys and would like to see even more practice sharing and information on financial data and new customer cases. They also appreciate sharing personal information. The weeklies were mentioned by the majority of the team members as a practice to be continued in 2018 as well.

All the answers above are in line with the outcome from the focus group. Based on the information above, the team members want to see the team developing towards a community more than a work team. The basis of this community is trust, transparency, collaboration and growth.

### 3.3.4 OfficeVibe analysis

While the results from inside the team can give valuable information, it is interesting to observe the change in relation to the environment where the team activates: the company. One way of assessment is through the employee engagement, which is a central topic in the challenges the employees face. The OfficeVibe analysis refers to analysing the employee engagement in the continuous development team in relation to the overall employee engagement in Futurice. OfficeVibe is a survey tool that sends different questions selected from a questions pool to different employees, and calculates the metrics based on the answers. It does not specify different questions for the continuous development team.

OfficeVibe is analysing employee engagement using the following metrics (their definition is according to OfficeVibe):

1. Personal growth: the level of autonomy the employees have, whether they are improving their skills and if they believe in the bigger purpose of their role
2. Recognition: the quality and frequency of recognition employees receive
3. Relationship with peers: trust, collaboration and communication between the peers
4. Happiness: employees' level of happiness and their satisfaction with their work-life balance
5. Satisfaction: how satisfied the employees are with their compensation and benefits, their role inside the organization as well as their overall work environment
6. Ambassadorship: level of pride the employees have towards the organization and if they would recommend it to other people
7. Feedback: the quality and frequency of feedback that employees receive as well as the consideration of their opinions and suggestions by the organization
8. Relationship with manager: trust, communication and collaboration between employees and their direct manager
9. Wellness: the level of stress employees feel at work and how they perceive the organization's efforts towards promoting healthy life habits
10. Alignment: how employees align themselves with the organization's vision, mission and values.

The metrics have values between 1 and 10, where 1 is minimum.

The continuous development team functions as a separate department in the company, and while it is not isolated from the company, it has a different type of customer work and different relationships within the customer cases. Based on the previous chapters, the team has a different approach to work and slightly different processes. Thus, comparing the OfficeVibe results for the continuous development team and the ones for the company could show some correlation between the different way of working and the employee engagement. The results below are taken on 2018 February 20<sup>th</sup>:

	Personal growth	Recognition	Relationship with peers	Happiness	Satisfaction	Ambassadorship	Feedback	Relationship with manager	Wellness	Alignment	Average
Futurice	7.80	7.50	8.10	7.20	7.50	8.30	8.20	7.80	6.40	7.80	7.66
Continuous development team	8.00	7.20	8.80	7.60	8.80	8.40	9.00	7.20	7.00	8.90	8.09
Difference	2.50%	-4.17%	7.95%	5.26%	14.77%	1.19%	8.89%	-8.33%	8.57%	12.36%	5.32%

Figure 59. OfficeVibe results, Futurice vs continuous development team. (Tauciuc, 2018)

The figure above shows that the continuous development team has generally higher scores than the overall company, with some scores higher. The only scores that are lower are:

1. Recognition: this is one area that should be improved, as to be at least at the same level as is in the company. The company provides different mechanism for recognition, and the team should be encouraged to take them into use. However, the continuous development team members feel however that they receive frequent and of quality feedback. This situation could also be tackled by repeating periodically the “a star and a wish” experiment.
2. Relationship with manager: at the time of the study, the supervisors of the continuous development team were not part of the continuous development team. Thus, the employees may feel disconnected from their supervisors, as the supervisors are not part of the daily work. However, there are currently plans to improve this situation.

All the other areas show a difference of minimum 1.19% (ambassadorship) and maximum of 14.77% (satisfaction). Based on that, we can conclude that the experiments had a positive impact in increasing the employee engagement in the continuous development team.

The top three scores are observed for:

1. Feedback 9/10
2. Alignment 8.90/10
3. Relationship with peers, Satisfaction 8.80/10

Based on the top scores, we can conclude that the continuous development team members believe in the company vision and see themselves as a part of it. They see their work as being important and meaningful in the context of the organization. They also have a good relationship with their peers, based on communication, trust and collaboration, and receive enough feedback.

The lowest three scores are related to:

1. Wellness 7/10
2. Recognition, Relationship with manager 7.20/10
3. Happiness 7.60/10

Out of these, the recognition and relationship with manager are lower than in the company. These areas need careful attention, and they were mentioned in a previous chapter. The other two metrics, wellness and happiness, are higher than overall in Futurice.

### **3.3.5 How relevant has the study been to the research objectives?**

The previous chapters of this section look at the impact of the experiments in two contexts: inside the continuous development team, and in the relationship between the continuous development team and the company. Based on the previous analysis, we can conclude that the experiments have had a positive impact. The purpose of the study was two-

fold: to provide a change inside the company, and to fulfil the research objectives mentioned in chapter 1. Below is an analysis on how this was accomplished for each objective:

**1. Understand what are some of the big challenges that the teams doing continuous development are currently facing**

Based on the theoretical framework used, team feedback and daily observations in the author's three roles, the big challenges have been identified in the following three areas:

1. working with the customers - having a mutual understanding of what continuous development means, having a shared direction of work
2. visibility of work progress – team's flow being interrupted, and the progress slowed down by customers constantly asking when they can expect certain features
3. team collaboration – transparency between the projects (as the team is sharing projects, one person in more projects, one project worked upon by more people), cultivating a culture of trust and collaboration, knowledge sharing

**2. Investigate how these challenges could be tackled with a LSC approach**

Based on the chapters 2 and 3 of this report, the challenges found in the continuous development are similar to the challenges that the teams implementing new services face. Thus, the *Premises for the study* chapter establishes that an LSC approach can be used to tackle the challenges. As soon as the problem areas have been identified, the study used the information from the continuous development retrospectives, as well as existing agile development tools, employee engagement approaches and principles from the LSC mindset in order to find a customized approach for the problems to be solved.

During the study, each challenge was approached with different LSC-inspired experiments. All the experiments can be considered as having a LSC approach (Nevanlinna, 2018).

**3. Create tools to implement the suggested solutions, possibly adding them to the LSC toolbox as extensions**

The study presents a series of experiments that touch on cultural alignment, mindset changes, spreading of existing practices from other sides of the company or embodiment of existing or prototyped new tools. The tools created in this study are both physical tools (e.g. canvases, time radiator), as well as practices (e.g. working in sprints, using story points, cross-teams dailies and weeklies) and combinations between practices and tools (e.g. "a star and a wish"). The new tools created during this study are:

1. The business objective canvas: it is an existing LSC tool that has been adjusted for the continuous development team's situation. The tool can be found in the company's common cloud storage solution and it can be used freely.
2. The focus canvas: is a new LSC tool inspired by the existing *weekly LSC board* canvas. The canvas is saved at the same location as the business objective canvas from above.

3. The time radiator, bringing transparency in how the time allocated per customer case has been used and helping with prioritizing work. Other teams inside the company expressed interest in using the time radiator for their own purposes.

#### 4. **Validate the above-mentioned solutions**

The study was started in spring 2017, and ended in December 2017, with the last assessments done during January 2018. The experiments mentioned as investigations of the LSC approach on the problem areas have all been carried on during the study period. Some experiments have been running a limited number of times (e.g. the canvases, the service manager as part of the development team, “a star and a wish”), others have been running periodically on a daily or weekly basis (e.g. working in sprints, usage of story points, cross-teams dailies, weeklies), while others have been running continuously (e.g. the time radiator). Similarly, some experiments have been validated for certain customer cases only (e.g. the canvases, the story points), while others have been validated across the team, regardless of the customer case (e.g. weeklies). The information from these validations has been discussed in the previous chapters.

#### 5. **Provide an approach that can help sustain innovation over time**

Based on the survey and team focus results, the study brought not only new tools, but a change in mindset. While the tools may change, the LSC mindset is geared towards continuous improvement that, in turn, sustains innovation. As seen in the previous chapters, the impacts can be seen at the individual level, as well as in the relationship between the continuous development team and the company, and in the way the team members report themselves to the company’s vision. Based on these elements, we can conclude that the study has used a strategic design approach. This means that the study is geared towards having a long-lasting impact, sustaining innovation over time, as it touches the organizations at three levels: at the individual level, at the networks level and at the frameworks, processes and resources level (Nixon, 2016):

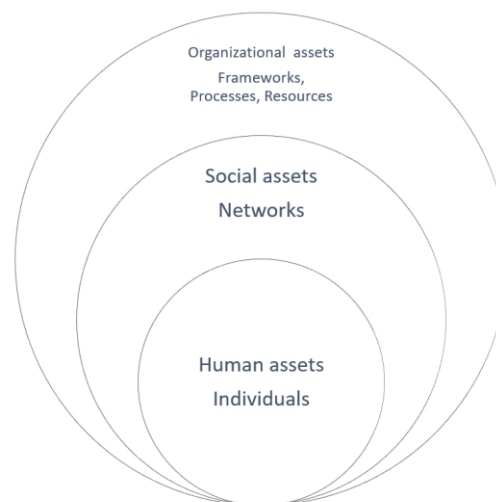


Figure 60. How strategic design impacts organizations. (Nixon, 2016)

## 4 Results

Based on the previous chapters, the study has had a significant impact on the continuous development team, and it has brought long lasting changes. The survey, focus group, observations and cognitive mapping suggest strongly that the experiments have brought an improvement in each of the problem areas tackled. An overview of the experiments can be seen below, together with the problem area they addressed:

Problem area	Experiment
Work with customer	Business objective canvas
	Focus canvas
	Service manager in case team
Visibility of work progress	Story points, velocity
	Work in sprints
Team collaboration and cohesion	Time radiator
	Cross-teams daily scrums
	Whole team weeklies
	"A star and a wish"

Figure 61. Practical applications of LSC approach for continuous software development problem areas. (Tauciuc, 2018)

The following sections discuss the impact of the experiments on each of the problem areas.

### 4.1 Work with customers

The main challenges about working with customers were identified as having a mutual understanding of what continuous development means and having a shared direction of work. Based on the team's feedback, the business objective and the focus canvases have proved effective in this area. The team members mentioned that the canvases "clarified business situation" and "they supported the clients to think over". Also, having the service manager as part of the team enabled a tighter feedback loop between the customer and the team. In the same time, this role added to the workload of the team and some comments from the team suggest that it may benefit more the customer to have a service manager with business expertise, outside the development team. However, this aspect could be discussed together with the customer, as his point of view is not known. The canvases will most probably be used again in customer cases, and the team will implement periodical retrospectives within the customer relationship.

## **4.2 Visibility of work progress**

Another type of challenges has been around the visibility of work progress. In practice, the following issues were identified: team's flow being interrupted, and the progress slowed down by customers constantly asking when they can expect certain features. The underlying problem was the lack of transparency of work progress towards the customer. This problem has been addressed by adopting a more agile way of working, including working in sprints and the usage of story points and velocity. While there were mixed results about the usefulness of working in traditional sprints, the team agreed that they bring more structure to the work, make it easier to plan the work and keep a healthy balance of the tasks. Also, the sprints have helped setting the work goals for two-weeks increments of time and communicating this to the customer decreased the number of interruptions motivated by lack of information. The story points have also had a positive impact, as they sustained the planning of the work. The story points have been visible for the customers as well and measuring the pace of work brought peace of mind for both the team and the customers by providing more accurate estimations.

## **4.3 Team collaboration and cohesion**

The last problem area addressed during this study is related to the team collaboration and cohesion. The problems belonging to this category have been: transparency between the projects, cultivating a culture of trust and collaboration, knowledge sharing. Following experiments were run: time radiator, cross-teams daily scrums, whole team weeklies and "a star and a wish". Based on the information received during data analysis, the team members have been very enthusiastic about these practices and they would like to continue them. The team members would like to have a stronger feeling of togetherness in the team – this need is based on the feeling that there is not enough knowledge sharing. In the same time, the team members appreciated the dailies and weeklies and suggested new ways on how to enable more sharing, for example by allowing more time flexibility and expanding the dailies concept to other expertise. They also suggested the team days as very important practices to continue and underlined the appreciation for personal information sharing during the weeklies.

OfficeVibe results also give an insight on the impact. By looking at the scores for relationship with peers, happiness, ambassadorship, feedback and wellness, we can see that the continuous development team members show a higher score than the general average in the company. Based on this, we can conclude that the experiments have brought closer



collaboration and team cohesion inside the continuous development team compared to other project teams in the company:

	Relationship with peers	Happiness	Ambassadorship	Feedback	Wellness	Average
Futurice	8.10	7.20	8.30	8.20	6.40	7.64
Continuous development team	8.80	7.60	8.40	9.00	7.00	8.16

Figure 62. OfficeVibe metrics influenced by team collaboration and cohesion, February 20<sup>th</sup>, 2018. (Tauciuc, 2018)

## 5 Conclusions

This study report has covered the research question and objectives, the theoretical framework used as a basis for the research, has uncovered the problem areas, suggested ways of alleviating these problems, validated these ways and discussed what their impact for the customers of the study was (the continuous development team). The *Results* chapter emphasized that the experiments ran during this study addressed the right problems and had a positive and long-lasting impact on the way of working.

Based on the author's observation and cognitive mapping during discussions with different stakeholders (sales people, other Futurice employees, account owners, business leads), the experiments helped change the perception of the team inside the company. The team became more aligned to the Futurice's philosophy and LSC way of working, and it has started being seen as a partner in developing the services, rather than the traditional view of a team that does maintenance. Futurice employees became interested in becoming part of the team, one even mentioning that "in this team I have seen a higher team spirit than in some of the customer projects where I worked". Of course, part of the reason is also the permanent setup of the continuous development team, compared to the average six months duration of a project. The team earned a more visible role in the company, got autonomy over the profit and loss and received strong support for growth. All these are partly side effects of the experiments being run.

### 5.1 Current state

Based on the information above, we can conclude that the LSC tools can help the developers involved in continuous software development by offering a platform of innovation and continuous improvement and diminishing the gap between the perceptions of creating new services and continuously developing existing services.

The initial question that the study aimed to tackle was:

How can LSC tools help developers overcome their problems during the software continuous development phase?

During the study, it was observed that the tools have influenced the mindset of the team members, thus creating a ripe environment for the build-measure-learn cycle. LSC tools are not about extending a canvas to the full service life cycle flow, but they are about the

process (Nevanlinna, 2018). Thus, the LSC tools brought changes in the process of perceiving, talking about and handling the continuous development.

## 5.2 Recommended next steps

As shown throughout this report, the study is always evolving and in order to get better results, the build-measure-learn mentality needs to be applied. The experiments applied in this study could be iterated and adjusted more.

The data analysis uncovered several areas that should be addressed further, such as: how to provide more time flexibility to the team, how to increase knowledge sharing, how to organize the service manager role in a better way, how to provide the best balance between challenges and time to learn. The team already offered suggestion to some of these areas, and it is recommended that these suggested should be experimented and adjusted so as to serve the team.

On a broader level, another aspect that should be addressed is changing the view from the continuous development team to the continuous development service. In practice, it means switching the mindset of the team towards creating and developing a service – the team already hinted at this in the part 2 of January 2018 survey. For this approach, the desirability of the service needs to be assessed, as desirability fires desire in customer. “Desirable interactions are something you tell others about and which give trust with, and loyalty to, the service”. (Stickdorn & Schneider, 2012). As customers, we need to consider all players from the stakeholder map shown in a previous chapter. The map needs to be reviewed and updated:

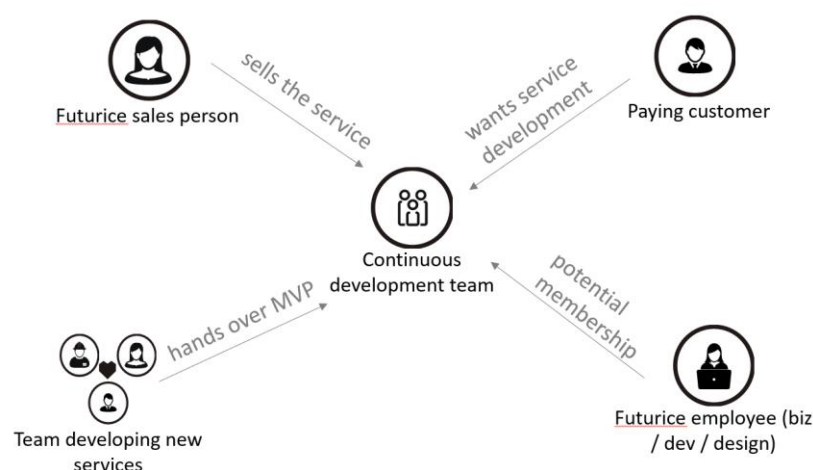


Figure 63. Continuous development team stakeholder map. (Tauciuc, 2018)

To make studying the desirability more tangible, Stickdorn & Schneider identify the following components:

1. Utility – what the service does, or offers to the customer, at the functional level
2. Usability – how easy it is to interact with the service
3. Pleasurability – how pleasurable the interaction is at the emotional level

Different tools can be used to assess and develop the service, such as customer journeys, use cases, challenging assumptions, personas and product – market fit analysis. Throughout the service development it is recommended to use a designing for growth approach, that minimizes the growth pains. (Ingle, 2013).

## **6 Discussion**

The findings from this study are universal and can be applied internationally: just like LSC, they are not dependant on the type of service, type of customer, location or size of work. They can be applied to any team that is doing continuous development: any team that is developing a service already launched. Moreover, they can also be applied regardless of the business sector, to any team who doesn't yet have many LSC principles in place, or who wants to have a better and smoother cooperation inside the team and with other stakeholders, regardless of the team's work objective.

The results of the study are also influenced by the three roles the author has had, as continuous development service lead, service manager in customer cases and researcher. Using daily observations, shadowing, having several discussions with different stakeholders helped shape the experiments and steer them towards the desired results. Also, as having ownership over the profit and loss of the team, the author's interests may have come in conflict with the team's interests, for example when prioritizing customer work over practice sharing. These aspects have been observed during the focus group and in the survey answers.

Looking at how the continuous development team functions now, we can observe the following characteristics:

1. It is based on democratic principles
2. It enabled social transformation
3. It values education
4. It is organized as a collaboration between micro teams

The same characteristics are mentioned by Hock when discussing Mondragon Corporacion Cooperativa in the Basque county in Northern Spain as an example of a chaordic organization. It began in 1956 as a tiny factory manufacturing paraffin stoves, it has grown

into a leading industrial group in the Basque region and seventh largest in Spain, with sales of 9.655 million euros in 2003. This is one example of the power of chaordic organizations. Following the same principles, the continuous development team could develop into a powerful team, that can grow into a very stable and profitable entity. However, this is only one direction that it can take.

The continuous development team is, at the time of writing this report, acting as a separate entity inside the company. However, since the study proves that the work the team does is very similar to the pre-MVP launch work, it is worth investigating whether another arrangement would be better suited for the team and the customers. The declared scope of the team is to grow the services, and this should reflect more in the services offered by the team. Throughout the duration of the study, the team's offer has been iterated many times, and the experiments, especially the canvases, provided helpful material in shaping the offers. Thus, while the study focused on three problem areas, it is important to adopt a broader view and put the findings in a more holistic context. It is also important to understand that any change takes time and using a MVP mentality brought quick feedback about the choice of experiments.

Based on these two directions, the team could benefit from deciding which option to focus on and use the available resources for building a strong service that fits the vision.

## References

- Berkun, S. 2005. Art of project management, O'Reilly.
- Business goes Social website, URL <http://businessgoessocial.net/blog/frederic-laloux-rsa-talk-reinveinting-organizations-cartoons/> , accessed 8.2.2018
- Cohn, M. 2006. Agile estimating and planning. Pearson Education, Inc.
- Collins, R. 2016. Reinventing management, Part 1: What colour is your organization?, URL [https://www.huffingtonpost.com/great-work-cultures/reinventing-management-pa\\_b\\_9387286.html](https://www.huffingtonpost.com/great-work-cultures/reinventing-management-pa_b_9387286.html) , accessed 8.2.2018
- Futurice website, URL <https://futurice.com> , accessed 5.2.2018
- Gothelf, J. & Seiden, J. 2017. Sense & Respond. Harvard Business Review Press.
- Grubb, P. & Takang, A. A. 2003. Software maintenance: concepts and practice (Second edition). World Scientific.
- Hock, D. 2005. One from Many. VISA and the rise of chaordic organization. Berrett-Koehler Publishers, Inc.
- Hüttermann, M. 2012. Agile ALM. Lightweight tools and agile strategies. Manning Publications Co.
- IDEO website, URL <https://www.ideo.com/>, accessed 7.2.2018
- Interaction Design Foundation website, URL <https://www.interaction-design.org/literature/topics/focus-groups> , accessed 20.2.2018
- Klein, L. 2013. UX for Lean startups. Faster, smarter user experience research design. O'Reilly Media, Inc.
- Laloux, F. 2014. Reinventing organizations. Nelson Parker.
- Lean Enterprise Institute website, URL <https://www.lean.org/> , accessed 7.2.2018
- Lean Service Creation website, URL <https://leanservicecreation.com/> , accessed 5.2.2018
- Leffingwell, D. 2007. Scaling software agility. Best practices for large enterprises. Pearson Education, Inc.
- Liedtka, J. & Ogilvie, T. 2011. Designing for growth: a design thinking tool kit for managers. Columbia Business School Publishing.
- Macaulay, L. & Miles, I. & Wilby, J. & Tan, Y. L. & Zhao, L. 2012. Case studies in service innovation. Springer Science+Business Media.
- Martin, B. & Hanington, B. 2012. Universal methods of design. 100 ways to research complex problems, develop innovative ideas, and design effective solutions. Rockport Publishers.
- Martin, R. 2009. The design of business. Why design thinking is the next competitive advantage. Harvard Business Press.

Martin, R. C. & Newkirk, J.W. & Koss, R.S. 2012. Agile software development. Principles, patterns and practices. Pearson Education, Inc.

Maslow hierarchy of needs and employment engagement, URL <http://www.loyaltyworks.com/news-and-views/uncategorized/maslows-hierarchy-of-needs-and-employee-engagement/> , accessed 5.2.108

McDermid, J. A. 1991. Software engineer's reference book. Elsevier Ltd.

Medium website, URL <https://medium.com/why-it-is-time-to-re-design-company-structures/2-evolutionary-context-of-organisational-structures-dddaaa650ab5> , accessed 15.2.2018

Modig, N. & Åhlström, P. 2013. This is lean. Resolving the efficiency paradox. Rheologica AB.

Moreira, M. E. 2017. The agile enterprise. Building and running Agile organizations. Apress.

New trends in management website, URL <https://newtrendsmanagement.wikispaces.com/Knowledge+Conversion> , accessed 15.2.2018

Nixon, N. W. 2016. Strategic design thinking. Innovation in products, services, experiences and beyond. Bloomsbury.

OfficeVibe website, URL <https://www.officevibe.com/> , accessed 5.3.2018

Poppendieck, M. & Poppendieck, T. 2014. The Lean mindset. Addison-Wesley.

Project management institute. 2017. Agile practice guide. Project management institute, Inc.

Ries, E. 2011. The Lean Startup. Portfolio Penguin.

Rudkin Ingle, B. 2013. Design thinking for entrepreneurs and small businesses. Putting the power of design to work. Apress.

Schwaber, K. & Beedle, M. 2002. Agile Software Development with Scrum. Prentice-Hall, Inc.

Scrum Alliance website, URL <https://www.scrumalliance.org/why-scrum> , accessed 7.2.2018

Service design principles, URL <http://matthewtyas.com/service-design-principles/> , accessed 7.2.2018

Sinek, S. 2011. Start with Why: how great leaders inspire everyone to take action. Portfolio/Penguin.

Stickdorn, M. & Schneider, J. & co-authors. 2010. This is service design thinking. Basics – Tools – Cases. BIS Publishers.

The agile manifesto, URL <http://agilemanifesto.org/> , accessed 2.1.2018

The design squiggle, URL <http://cargocollective.com/central/The-Design-Squiggle> , accessed 7.2.2018

The designorate website, URL <http://www.designorate.com> , accessed 7.2.2018

The difference between design thinking, lean startup and agile, URL

<https://medium.com/@SteveGlaveski/the-difference-between-design-thinking-lean-startup-and-agile-5cf07b117562> , accessed 8.2.108

The Disciplined Agile Consortium website, URL

<http://www.disciplinedagiledelivery.com/teal-is-the-new-black/> , accessed 8.2.2018

The ExperienceUX website, URL [http://www.experienceux.co.uk/faqs/what-is-](http://www.experienceux.co.uk/faqs/what-is-ethnography-research/)

[ethnography-research/](http://www.experienceux.co.uk/faqs/what-is-ethnography-research/) , accessed 14.2.2018

The Lean Startup website, URL <http://theleanstartup.com/>, accessed 24.1.2018

The Scrum concept, URL

[https://upload.wikimedia.org/wikipedia/commons/thumb/c/cd/ScrumSchwaberBeedle.s](https://upload.wikimedia.org/wikipedia/commons/thumb/c/cd/ScrumSchwaberBeedle.svg/2000px-ScrumSchwaberBeedle.svg.png)  
[vg/2000px-ScrumSchwaberBeedle.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/c/cd/ScrumSchwaberBeedle.svg/2000px-ScrumSchwaberBeedle.svg.png) , accessed 6.2.2018

Womack, J. P. & Jones, D. T. 2003. Lean Thinking, 2nd Edition. Simon & Schuster, Inc.



## Appendices

### Appendix 1

### Agile Manifesto

The following principles of Agile Manifesto are transcribed from <http://agilemanifesto.org/principles.html>.

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Build projects around motivated individuals.

Give them the environment and support they need and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development.

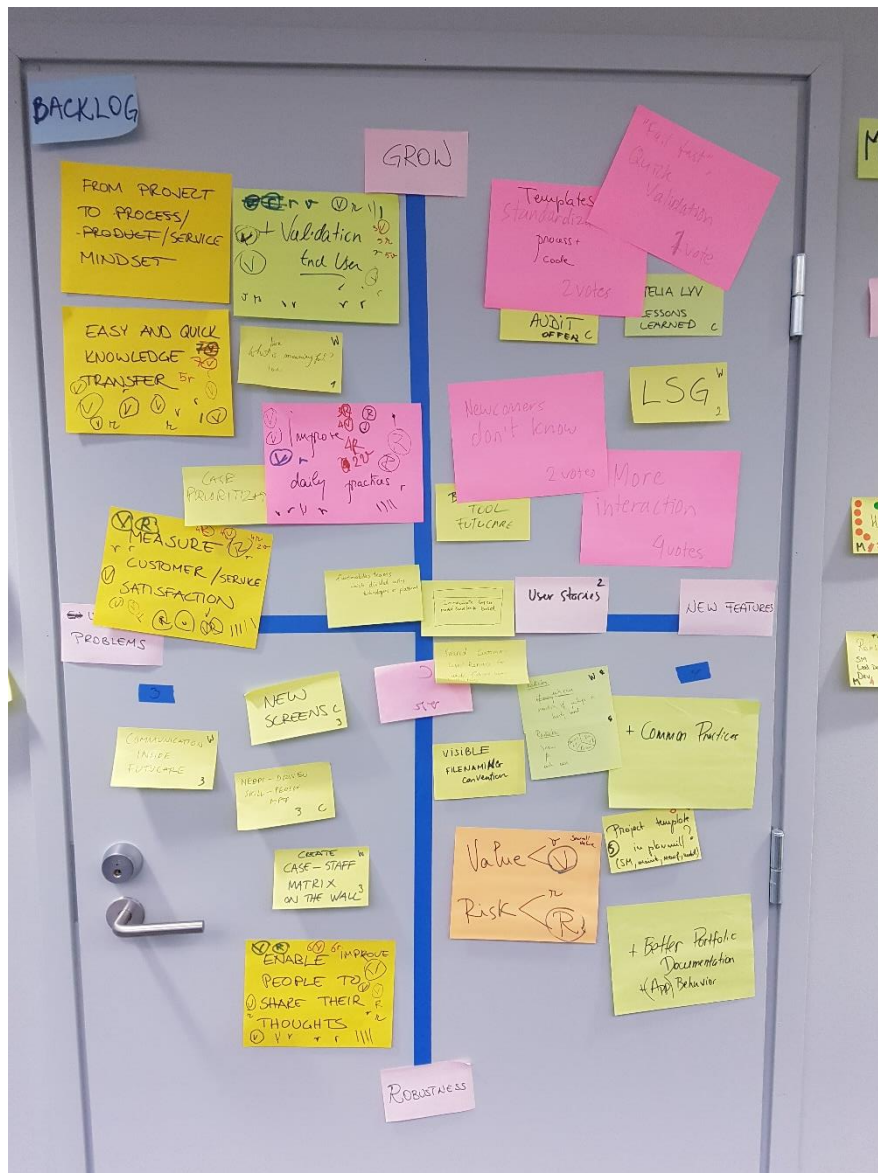
The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Continuous attention to technical excellence and good design enhances agility.

Simplicity--the art of maximizing the amount of work not done--is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.





**Warm up**

Introduce interviewer

Introduce the reason and theme of the interview

Explain how the answers will be used, by whom

Start with warming up the atmosphere

**The beginning of LSC**

What is the story behind the birth of LSC? Why did you create it?

What is LSC?

What does LSC stand for? What are its values?

**LSC vs its building blocks**

What is the difference between LSC, lean startup, design thinking and agile development?

What is the contribution of LSC?

**LSC content**

What does LSC consist of?

What role do the canvases have in the big picture?

**LSC and continuous development**

How may continuous development and software maintenance be part of LSC?

What are the current practices for the post-MVP phase?

What would you consider as LSC experiments? Are the experiments in this study LSC?

**Future direction of LSC**

What do you see as the future direction of LSC?

What role could LSC play in Futurice's 2023 vision?

If you had to start from the beginning, knowing what you know now, what would you do differently?

The structure used for the interview has loosely been the one above. The focus was on having a conversation that evolves naturally, guided by the main purpose of understanding the ideas behind LSC and the connection it has with continuous development in the context of the study.

Hanno Nevanlinna is one of the founders of Futurice. Currently, he is the director of culture and service innovation coach inside the company.

The interview took place on 30.1.2018. Following are notes taken by the author from the interview.

### The beginning of LSC

1. First, there was an initiative to change how the company was selling and doing projects. Design, business and tech skilled employees worked together and designed the first LSC process, as below:



Figure 64. LSC process (Futurice, 2013)

2. Next, as per request from a customer, the first toolbox was put together (lectures and workshops around lean, design thinking and agile philosophy).
3. The first canvases were created next, to bring more tangibility and order to the process.
4. The order of invention of LSC: the mindset, then the process, then the canvases.
5. Situation now: LSC it's spreading. Why? It has a rather strict workflow that creates trust, helps team focus on right topics, rather concrete process (do not need to read a book to be able to run it, just take the canvases), easy enough to take into use in different type of organizations and challenges.
6. The values of LSC: human centricity regarding the team work, customer centricity, transparency, continuous learning, high-level and detail-level focus.

### LSC vs its building blocks

1. LSC brings the main tools from the 3 different schools to one single process, so one team can tackle all
2. It aims to have a common language throughout the whole creation process. The 3 schools of thought have already a lot in common, but none of them tackle all of the main points, for e.g.:
  - Agile has boxed itself as a software development process, not about creating the business or service
  - Design Thinking is way too much about the user centricity, way too often forgetting that's about the business we can do by serving the business needs. Designers interact with designers only.

- Lean startup doesn't tackle the building side of creating a service, or how will we understand the user needs. However, it gives a good framework on how to iterate the whole concept.
- 3. LSC builds on top of these concepts, providing a common language so that people with different expertise can communicate effectively and work together. It focuses on team work.
- 4. Current tools are not evolving (e.g. the business model canvas) or are difficult to fill in and counterintuitive (e.g. the value canvas that needs to be filled in from right to left, unlike the established way of writing from left to right, bringing confusion).

### **LSC content**

1. LSC consists of the mindset, the process and the tools (the canvases)
2. Why canvases? Canvases are making LSC complete. How?
  - Hanno believes in physical team work. When team sits at the same table, the opinions come forward easier. In trello for e.g. it can get easy in the situation when someone is active and dictates the whole process.
  - Hanno loves the idea of communicating with the walls (concept coming from agile). "Years ago, you were not allowed to touch the walls in some customer premises. I wanted to change that." (Nevanlinna, 2018)
  - Canvases are a tool to ask the right questions. They are not a documentation tool.
3. Most of the applications of LSC are done with the original set of canvases. Some translated them to their own language, some created add-ons and some are branding them to make them more company specific or are holding workshops on them.

### **LSC and continuous development**

1. When talking about fixing bugs, LSC cannot help. However, LSC is valuable when the aim is to grow the service and make it more relevant to its users. "Data is the king when the service is launched".
2. It's also related to the size of the change:
  - If a change is small, just test it out
  - The bigger the change is, the more you need to talk to the customer about it, the more you would benefit from taking more canvases into use, as there is a clear business goal.
3. Current post-MVP practices in Futurice:
  - The idea of service creation is that you build, then you measure, then you learn
  - Customers are used to buying MVPs – they believe that when it's done, it's done and ready. The scope of the next iteration of LSC (LSC fullstack) is to tackle the company internal approach of stopping at the MVP
4. What are considered as LSC experiments:
  - Any experiments related to the LSC framework
  - It's not about extending a canvas to the flow, it's about the whole process
  - LSC is a well created set of tools that serve the process
  - In this light, the experiments in this study can be considered LSC

### **Future direction of LSC**

1. Next iteration of LSC, called LSC fullstack, bringing more tools around team work and the process
2. The value of LSC stands in it being compact, tangible, easy to understand, easy to teach, and working in different contexts (digital, non-digital). We need to keep that value really high. It can't grow forever as a toolset. It can evolve forever, but it shouldn't grow.
3. Knowing everything I know now, I will do LSC the same way.



## Appendix 6

## Team survey invitation

Invitation to fill in the survey on 2017 sent to the continuous development team. Invitation was sent via email and flowdock on 25.1.2018.

Hi,

### **in short:**

You are the ones building Futucare.

Jan 2018 pulse + 2017 retro survey: <https://goo.gl/forms/BBAwvoU2QTTyVxhx2>

Please reply by end of Friday 26.1.

Results will be used as recommendations for new Futucare lead + in my thesis.

### **long version:**

Your ideas and your work are what make this team going forward. Because a new chapter is starting, let's use the 2017 retro to make it a great chapter.

We had some insights last week in the team day, and I would like to get a bit more practical as well.

So, please reserve about **30 mins to answer this modified version of the**

**pulse:** <https://goo.gl/forms/BBAwvoU2QTTyVxhx2>. It contains the pulse questions and questions about 2017. Many of them are open questions, so it's your chance to bring your ideas forward into building Futucare 2018.

Could you fill this in **by end of Friday 26.1**? It goes nicely with a cup of coffee or tea :) I will use the results as suggestions for new Futucare lead, and also in my thesis. Big thank you!

If everyone answers by end of Friday, I will bring a cake on Monday.

Loredana



The questions below refer to “Futucare”, as the internal name for the continuous development team. “Futucarist” is a person part of the Futucare team.

1. I can rely on my colleagues’ skills and abilities
2. I feel I can honestly discuss my tasks, even negative feelings and frustration
3. I get support from my colleagues when I need it
4. I do have enough visibility and information about what is going on in the Futucare team
5. I think that we share enough practices between cases
6. I have challenged myself (tried new things, learned new things, shared my knowledge etc.) recently
7. I feel that my workload is in balance?
8. I think that my current project/work tasks are interesting
9. I am proud of being a Futucarist
10. I get enough feedback about my work
11. I feel like I can influence things at work to improve things around me
12. I feel like I'm very good at my job
13. I feel that I have possibilities to improve my professional skills in my work
14. We have a strong sense of togetherness in Futucare team
15. I enjoy working at Futucare
16. I would recommend Futucare team for other Futuricians
17. I feel that my feedback is taken into account well in Futucare
18. I feel we should do these pulses regularly

## Appendix 8

## Team survey questions, August 2017

The questions below refer to “Futucare”, as the internal name for the continuous development team. “Futucarist” is a person part of the Futucare team.

1. I can rely on my colleagues’ skills and abilities
2. I feel I can honestly discuss my tasks, even negative feelings and frustration
3. I get support from my colleagues when I need it
4. I do have enough visibility and information about what is going on in the Futucare team
5. I think that we share enough practices between cases
6. I have challenged myself (tried new things, learned new things, shared my knowledge etc.) recently
7. I feel that my workload is in balance?
8. I think that my current project/work tasks are interesting
9. I am proud of being a Futucarist
10. I get enough feedback about my work
11. I feel like I can influence things at work to improve things around me
12. I feel like I’m very good at my job
13. I feel that I have possibilities to improve my professional skills in my work
14. We have a strong sense of togetherness in Futucare team
15. I enjoy working at Futucare
16. I would recommend Futucare team for other Futuricians
17. I feel that my feedback is taken into account well in Futucare
18. I could help organizing team days

## Appendix 9

## Team survey questions, January 2018

The questions below refer to “Futucare”, as the internal name for the continuous development team. “Futucarist” is a person part of the Futucare team.

Part 1, questions common to the surveys from June 2017 and August 2017:

1. I can rely on my colleagues' skills and abilities
2. I feel I can honestly discuss my tasks, even negative feelings and frustration
3. I get support from my colleagues when I need it
4. I do have enough visibility and information about what is going on in the Futucare team
5. I think that we share enough practices between cases
6. I have challenged myself (tried new things, learned new things, shared my knowledge etc.) recently
7. I feel that my workload is in balance?
8. I think that my current project/work tasks are interesting
9. I am proud of being a Futucarist
10. I get enough feedback about my work
11. I feel like I can influence things at work to improve things around me
12. I feel like I'm very good at my job
13. I feel that I have possibilities to improve my professional skills in my work
14. We have a strong sense of togetherness in Futucare team
15. I enjoy working at Futucare
16. I would recommend Futucare team for other Futuricians
17. I feel that my feedback is taken into account well in Futucare
18. I could help organizing team days

Part 2, questions about 2017 retrospective:

19. What would be ideal the ideal content for a team day?
20. What stops us doing our daily work better and how would you fix it?
21. How would you describe in 3 words Futucare's journey in 2017?
22. What practices/new ways of working did we try out in 2017?
23. What practice had the biggest impact on you? What was the impact? (positive or negative)
24. We have used LSC canvases in at least a customer meeting.
25. What impact did the canvas(es) have on the meeting/work/collaboration? (mention if you haven't used any)
26. What are the usual challenges you have in working with customers?
27. How might we make our meetings with the customer more effective? (e.g. kick off, status meetings etc.)
28. How do you see the role of service manager in the team? (sm is also a developer, sm is a business person from Futucare, sm is also the account owner, other?) Why?
29. I have used story points for our backlog items.
30. What impact did measuring velocity have on your project? (mention if you haven't measured it)
31. Are you working in sprints? What impact did it have on the work?
32. What has been the impact of using the radiator?
33. Are you aware of any project where someone from Futucare team worked before the project came to Futucare?

34. If you answered "yes" before, what was the impact of the team being involved in the project before the Futucare phase started?
35. Have you been involved in the android daily scrums? What was the impact of those chats?
36. Have you been involved in pair programming? What was the impact of this way of working?
37. Have you been involved in the "a star and a wish" pair feedback session? What was the impact of the session on you?
38. What kind of information would you like to see in the Futucare weeklies?
39. What do you tell about Futucare to Futurice people outside Futucare?
40. What would you like to see happening in 2018 in Futucare?
41. What practices from 2017 would you like to see continuing in 2018?
42. What new things should we try in 2018?

The following shows the detailed answers received for the questions from Part 2 of the team survey held in January 2018. Please see Appendix 9 for the questions.

19. What would be ideal the ideal content for a team day?

- Togetherness, stuff that affects everyone, not just few cases stealing the time.
- Retro when found necessary + how we work towards our goals in coming months + info of new customer cases and need of new developers + fun. Preferably half a day than one whole day & not too often
- Introducing Getting Things Done -system, every sharing architecture or 'devops' practices of their customer cases
- How to work better together?
- Leisure activity, tech activity, lunch
- Retro, selected operational improvement planning.
- Hackathon!
- I like casual hanging out together but would also like to do some knowledge sharing between case teams and tinker about continuous futucare improvement things.

20. What stops us doing our daily work better and how would you fix it?

- People are busy and are not that easily available to help when needed
- Lack of common shared practices testing, tools etc. Devops is a good start going forwards.
- Cases are still 'silos'. It's not easy to help others working on other cases due to complex environment setup, knowledge transfer needs etc.
- Little knowledge is shared between people in different projects but solving similar problems. More knowledge sharing sessions.
- Tech expertise missing some areas, learn it (devops, aws)
- Everyone is a generalist and need to "re-invent the wheel". Let people specialize on some areas and share this information with others when needed.
- We're always in a hurry to produce new features faster. Organize around competences to emphasize impact instead of feature throughput.
- I don't have the willpower to commit to push more iterative ways of working with the clients. Also, I think we should have some bench constantly in Futucare to be able to react to busy months caused by mini projects/new projects/package changes.

21. How would you describe in 3 words Futucare's journey in 2017?

- Very noticeable growth
- received organizational state
- Interesting and eventful
- Growth, profitability, fun
- challenged, busy, interesting
- Big time expanding
- Growth, brand, continuous.
- Established, accepted, expanded.

- teaming, communicating, doing
22. What practices/new ways of working did we try out in 2017?
- Cannot point a finger at anything
  - new location, some learning opportunities like aws
  - Team days, pomodoro system, weeklies, refreshment events
  - Daily, Pair coding, Primary-backup
  - Did we? Sorry could not remember new practices. Maybe because I started in 2017.
  - Dailies --> weeklies.
  - Internal and external salespersons.
  - We tried to help each other and as for help more. We sought issues that needed solving and tried to fix them or at least keep them in mind. We communicated a lot internally to raise awareness and to get more resources and clients. We hanged around together every week and moved to a space that could fit us all. We talked more about our work and tried to support each other. We figured out new things we could do that we saw demand for. We tried to figure out what kind of cases we should have and how to sell us to the company and to the clients. We started to expand to other sites.
23. What practice had the biggest impact on you? What was the impact? (positive or negative)
- no bigger positive or negative impacts
  - I don't think that we have that many practices, so I would say weeklies. A good way to share what other team members have been doing.
  - Weeklies: to know better what is going on cases and what is coming in
  - Primary-backup since it increases mobility between cases.
  - less overhead
  - Internal sales enabled us to hire one new developer-employee of our own!
  - For me teaming and growing was probably the most impactful things in a positive way.
24. We have used LSC canvases in at least a customer meeting.
- Don't remember
  - No
  - No
  - No
  - No
  - Don't remember
  - Don't remember
  - No
  - Yes
  - Yes
25. What impact did the canvas(es) have on the meeting/work/collaboration? (mention if you haven't used any)
- Never saw them used
  - haven't used any
  - Haven't used (started less than two months ago)
  - I don't think I remember any impact of the canvases or I don't believe much in the canvases in general.
  - Don't know the abbr. LSC, perhaps used it.

- not used
- Clarified business situation
- They supported the clients to think over

26. What are the usual challenges you have in working with customers?

- Communication between tech and not tech people
- Learning the new customer and its business and environment takes time
- Communication and conveying the importance of continuous support
- It's been really easy to work with the customer. Customer's tickets could be a bit better but it does have made my work any harder because it's customer answers questions quickly.
- Prioritization of tasks changing too often and seeming ad hoc. Not enough effort to long term planning
- Explain why it is taking time in the project.
- Customer has no power to make up decisions, which should speed up things at hand.
- There aren't any outstanding challenges
- Clients only buy features, not the things that make services work.
- Lack of time to prepare

27. How might we make our meetings with the customer more effective? (e.g. kick off, status meetings etc.)

- Someone should be actively keeping us off the deep details most of the time
- Regular meetings with agendas.
- We have a minimal number of customer meetings = once a week
- Fixed agenda and/or timeboxing (within meeting)
- Keep the meeting time box. Have predefined goals for the meetings. Clear agenda.
- Make things up before hand in presentable way.
- Be more brave to talk about impediments and risks and what kinds of work are required to face them.
- I would like to use LSC to drive focus on important matters in short and long term

28. How do you see the role of service manager in the team? (sm is also a developer, sm is a business person from Futucare, sm is also the account owner, other?) Why?

- Interface between clients, team and the business people. The team will not have the time to know all the business stuff properly because client work takes precedence. And we may not be the best folks to be always the only contact point for the client. Sometimes the clients need someone more detached from the everyday work.
- Service manager can have different kinds of roles. It should bring value to customer relationship.
- Ideally, I would see sm as account owner. To know customer and customer systems. Authority to make decisions and protect developers
- I think a service manager is important in a case to keep the communication easy for both sides.
- Business person/AO. There are lot of projects and lot of different tech details, which cannot be easily tackled unless working on project in daily/weekly basis. So, unless SM is a super person, that's not humanly possible.
- There is no SM from Futucare in my customer case, but it would be good to have one.

- Continuous contact with client, feeling the pulse, finding and fixing things to handle around the case. Can be any other role as well but needs to understand actual work issues and provide guidance and insight. This needs time seldom left over from feature development, and more than an Account Owner can possibly have per client.
- I prefer to have someone in the team do it, but with some clients it would sink them. In those cases, we should lose the client or have a dedicated service manager

29. I have used story points for our backlog items.

- Yes
- No
- No
- Yes
- Yes
- Yes
- Yes
- Yes
- Yes
- No

30. What impact did measuring velocity have on your project? (mention if you haven't measured it)

- Cannot recall seeing velocity measured
- haven't measured
- Hard to say since have been working here for too short time
- We/customer doesn't measure it. Project is not a real scrum project.
- Not really measured
- It has positive impact if the velocity is not something countable.
- Project self-learning with measuring things and task are improving.
- None
- None.
- Haven't measured. We have tried to split tasks into really small ones though.

31. Are you working in sprints? What impact did it have on the work?

- Not really
- No
- Sprints aren't that useful when mainly responding to problems in services (which are unpredictable)
- Not really. We implement one task and move next one when it's ready.
- Yes, but only in very light-weight. Basically, for setting bi-weekly goals
- Yes. It good to keep the workload reasonable to the team.
- Yes, we are. They define a pace within two weeks.
- No
- Could've used Kanban with periodic checkpoints and much the same results.
- No sprints, more like release as often as possible

32. What has been the impact of using the radiator?

- It is a good reminder of how and where hours have gone. Or what has been ignored.
- No / minor impact in my customer case
- Easy to get knowledge on status of different cases on quick glance



- No
  - It's great to see overall status. Also, easy way to monitor status of my cases
  - It is important to keep track of cases.
  - When it's working then it creates visibility on the pace where hours are spent.
  - Not much, nice to see if hours are accumulating as expected.
  - When it works, it helps to choose what to work on, and provides a visual place for customer case to talk about.
  - I like to have it visible. And I think it has made a difference
33. Are you aware of any project where someone from Futucare team worked before the project came to Futucare?
- Yes
  - Yes
  - No
  - No
  - No
  - Yes
  - Yes
  - No
  - Yes
  - Yes
34. If you answered "yes" before, what was the impact of the team being involved in the project before the Futucare phase started?
- I really cannot say
  - Knowledge sharing, and learning shorten the time to success
  - Less time of knowledge transfer.
  - Greatly helping transition in terms of learning and also with customer relationships.
  - Continuation becomes almost too natural; care has to be taken to share the case and get others to work on as well. IDK really, but assume it helps to maintain client satisfaction during transition (since they almost don't see any).
  - One has a better understanding of the service, the domain and customer expectations. Also one can spot issues that don't fit into Futucare way of working and try to fix some of them before Futucare phase starts.
35. Have you been involved in the android daily scrums? What was the impact of those chats?
- No
  - No
  - No
  - We used to have those around our customer case. It was ok way to know on what features others were working on and hear if anyone need feedback/review
  - Yes. Reveal the commitments of the team to the case and the potential problems.
  - No
  - No
  - No. I enjoyed when people had those. Helped them start sharing more.
  - no

36. Have you been involved in pair programming? What was the impact of this way of working?
- Yes. It is very educative
  - No
  - Helps to get onboard the project faster and great way to learn fast from more experienced colleague
  - Sometimes. It's a good way figure out the problem. Ahaa-moment usually comes when you have to explain code to somebody else.
  - Only on knowledge transfers
  - Yes. It is really helpful for someone to get up to speed in a new case. In the current cases, it is help decreasing the technical gap between team members.
  - I have done pair programming, it helps when the task in hand is not totally clear and requires searching knowledge of architecture/techs. When the task is simple enough it consumes more time than doing it solo or separated.
  - No
  - It causes a lot of talk, and then very solid solutions emerge. Features become more useful and polished. Happiness increases. Satisfaction is expressed afterwards.
  - Yes. For me it's a good way to get into a codebase, because one can ask questions all the time from the more experienced one. It can also help the whole team understand a key feature better together than trying to split it into pieces and approach separately. But it's not for everyone as it can be stressful.
37. Have you been involved in the "a star and a wish" pair feedback session? What was the impact of the session on you?
- I guess I have. It was nice to hear all these things
  - No
  - No
  - Okish feeling about those. If pairs are selected by random then it may be difficult to find feedback if haven't been working closely. Cumbersome at least for the beginning. Could be that would become more smooth and relaxed over time.
  - Yes. Feel grateful to have the person by your side.
  - No
  - No
  - Very good discussion and formulation of needs. No further action to act on those needs though.
  - Don't remember that
38. What kind of information would you like to see in the Futucare weeklies?
- Less details
  - smileys, troubles, info of changes
  - Finance, new and ending projects, feedback from customers
  - Case statuses (shortly), new deals (with their tech stacks). Long term (=months) plans for Futucare
  - Smileys. Cases. Share of Practices.
  - Current weekly flow is mainly ok. Everybody gets a way / channel to express themselves.
  - Lead / business status. Situation in other projects and new technologies used.
  - History, future, profit, investment. Short demos of mad experiments!
  - Personal stuff, case updates and news, team related stuff

39. What do you tell about Futucare to Futurice people outside Futucare?

- What we do in short and how we have some longtime members and then some guest stars who do tours of duty and then go back to projects. Depends on who I talk to.
- Nothing much, but if I was asked I would say that it's a team full of friendly and skilled people
- We do long term support and continuous development while constantly improving service reliability and availability.
- I don't have that much connections within Futurice outside Futucare. Probably internals connect more due welcoming, mads and other company internal events. As external I don't feel I have luxury to just hang around and chat with people. Pressure to make billable work is probably more pressing than for internals.
- Great people.
- It's about development, maintenance and customer expectancy handling.
- I don't talk too much to other futurice people. If I do, then it is about some issue related to project.
- We are a permanent team that does small-scale development for selected long-term client cases.
- I tell them that the pace of working is different and although it involves more context switching, it's a good change after long time in projects. Learning is also broader instead of focused in projects and one can influence one's work a great deal. Also, one gets to spend more time at the office and with a broader set of people and can become more aware of the company-related stuff than at the client's.

40. What would you like to see happening in 2018 in Futucare?

- Devops, QA, more subteams and people getting to do different things
- Some new customer cases
- DevOps effort being successful in unifying tooling and ways of working across several customer cases (optimally all, but not really possible because of the difference in technologies)
- New interesting projects / technologies /
- Continuing growth, birth of DevOps, harmonizing ways of working (for development environments, git workflows, docking, ...)
- More time invest in learning from experienced people.
- More customer mini projects or participating in Futucare client projects as they help with the actual customer maintenance work. Time for devops inside futucare.
- More co-operation between projects.
- Our needs met! COURAGEOUS RECRUITMENT! Teaming up! Experiments! DevOps&QA sales like mad!
- Continuing futucare improvements, sharing knowledge within the team and doing good work with the clients. Committing to push more iterative way of working in cases.

41. What practices from 2017 would you like to see continuing in 2018?

- Subteams like the mobile team in the summer
- weeklies, team days
- Weeklies
- Weeklies (in one way or another), team days (with good preparation/planning & followup), refreshment/team building events
- Pair coding, daily, primary-backup.
- Weeklies, teamdays.

- Team days
- Everything we have now?

42. What new things should we try in 2018?

- More rotation of people between projects
- Devops / improving code analysis / tools / testing.
- I don't have anything specific in mind now. Some ways that would lower down the silos and make it easier for futurcarers to participate and help in other projects.
- Short knowledge sharing session.
- Some kind of project welcoming phase++. In complex enough projects throwing in the deep cold end is not enough.
- Small demonstrations on new things tried with projects.
- Sell experimentation and discovery of benefits to clients. Instead of a feature, make an experiment, every once in a while. Obviously, make quality and infrastructure work benefits more visible to clients as well.
- The iteration via practicing LSC with the client... Coaching new people to become Service Managers in one of their projects?

1. Comments related to 2017:
  - i. not able to concentrate - too much switching between tasks
  - ii. work is not always very meaningful, for e.g. like the chilicorn projects
  - iii. backup needs – some team members felt they get the support they need
  - iv. the current continuous development team is a bit abstract. we are small teams building a bigger team. we cannot help other teams because we don't know the context
    1. we are not a team. just a bunch of different teams
    2. what helped: the android dailies
    3. problem: not enough time set aside as investment in learning
  - v. failed promise: in the interview "you will work with a senior", but the senior hasn't had time to teach
  - vi. how do we keep up with new technology? our customers use old tech
    1. pull manner -> suggest to customer
    2. problems from outdated code -> idea: migrate the systems
  - vii. new team members:
    1. when I joined: what was promised - being a team (ok), big projects going on for years (ok), things we can improve: I am a full stack dev with focus on frontend and I cannot see a lot of frontend (how it was promised), we don't work with modern tech - can we make this into our business model?, get more projects to work on new tech; other solution: build part of the systems with new tech (digital transformation).
2. Decisions:
  1. periodical retros in every customer case.